



Guía del usuario

# Aplicación de escalado automático



# Aplicación de escalado automático: Guía del usuario

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

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

---

# Table of Contents

¿Qué es Auto Scaling de aplicaciones? .....	1
Características de Auto Scaling de aplicaciones .....	2
Trabajar con Application Auto Scaling .....	2
Conceptos .....	3
Más información .....	5
Servicios que se integran .....	6
Amazon AppStream 2.0 .....	8
Rol vinculado a servicio .....	9
Entidad de seguridad de servicio .....	9
Registro de flotas AppStream 2.0 como objetivos escalables con Application Auto Scaling .....	9
Recursos relacionados .....	10
Amazon Aurora .....	10
Rol vinculado a servicio .....	10
Entidad de seguridad de servicio .....	11
Registro de clústeres de Aurora DB como destinos escalables con Auto Scaling de aplicaciones .....	11
Recursos relacionados .....	12
Amazon Comprehend .....	12
Rol vinculado a servicio .....	12
Entidad de seguridad de servicio .....	13
Registro de recursos de Amazon Comprehend como destinos escalables con Auto Scaling de aplicaciones .....	13
Recursos relacionados .....	14
Amazon DynamoDB .....	15
Rol vinculado al servicio .....	15
Entidad de seguridad de servicio .....	15
Registro de recursos de DynamoDB como destinos escalables con Auto Scaling de aplicaciones .....	15
Recursos relacionados .....	18
Amazon ECS .....	18
Rol vinculado a servicio .....	18
Entidad de seguridad de servicio .....	19
Registro de servicios ECS como destinos escalables con Auto Scaling de aplicaciones .....	19
Recursos relacionados .....	20

Amazon ElastiCache .....	20
Rol vinculado al servicio .....	21
Entidad de seguridad de servicio .....	21
Registro de grupos de replicación ElastiCache (RedisOSS) como destinos escalables con Application Auto Scaling .....	21
Recursos relacionados .....	23
Amazon Keyspaces (para Apache Cassandra) .....	23
Rol vinculado a servicio .....	23
Entidad de seguridad de servicio .....	23
Registro de tablas de Amazon Keyspaces como destinos escalables con Auto Scaling de aplicaciones .....	24
Recursos relacionados .....	25
AWS Lambda .....	25
Rol vinculado a servicio .....	25
Entidad de seguridad de servicio .....	26
Registrar las funciones de Lambda como destinos escalables con Auto Scaling de aplicaciones .....	26
Recursos relacionados .....	27
Amazon Managed Streaming for Apache Kafka (MSK) .....	27
Rol vinculado a servicio .....	27
Entidad de seguridad de servicio .....	28
Registro del almacenamiento en clúster de Amazon MSK como destinos escalables con Auto Scaling de aplicaciones .....	28
Recursos relacionados .....	29
Amazon Neptune .....	29
Rol vinculado a servicio .....	29
Entidad de seguridad de servicio .....	30
Registro de clústeres de Neptune como destinos escalables con Auto Scaling de aplicaciones .....	30
Recursos relacionados .....	31
Amazon SageMaker .....	31
Rol vinculado a servicio .....	31
Entidad de seguridad de servicio .....	32
Registro de SageMaker variantes de puntos finales como objetivos escalables con Application Auto Scaling .....	32

Registro de la simultaneidad de puntos de conexión sin servidor como destinos escalables con Application Auto Scaling .....	33
Registro de clústeres de componentes de inferencia como destinos escalables con Application Auto Scaling .....	34
Recursos relacionados .....	35
Flota de spot (Amazon EC2) .....	35
Rol vinculado a servicio .....	35
Entidad de seguridad de servicio .....	36
Registro de flotas de spot como destinos escalables con Auto Scaling de aplicaciones .....	36
Recursos relacionados .....	37
Amazon WorkSpaces .....	37
Rol vinculado al servicio .....	37
Entidad de seguridad de servicio .....	38
Registro de WorkSpaces grupos como objetivos escalables con Application Auto Scaling .....	38
Recursos personalizados .....	39
Rol vinculado a servicio .....	39
Entidad de seguridad de servicio .....	39
Registro de recursos personalizados como destinos escalables con Auto Scaling de aplicaciones .....	40
Recursos relacionados .....	41
Configure el escalado mediante AWS CloudFormation .....	42
Application Auto Scaling y AWS CloudFormation plantillas .....	42
Fragmentos de ejemplo de plantilla .....	43
Más información sobre AWS CloudFormation .....	43
Escalado programado .....	44
Cómo funciona el escalado programado .....	45
Funcionamiento .....	45
Consideraciones .....	45
Comandos de uso frecuente .....	46
Recursos relacionados .....	47
Limitaciones .....	47
Cree acciones programadas .....	48
Creación de una acción programada que se produce solo una vez .....	48
Crear de una acción programada que se ejecuta en un intervalo recurrente .....	50
Creación de una acción programada que se ejecute en una programación recurrente .....	51
Crear una acción programada puntual que especifica una zona horaria .....	51

Creación de una acción programada recurrente que especifique una zona horaria .....	52
Describe el escalado programado .....	53
Describe las actividades de escalado de un servicio .....	53
Describe las acciones programadas para un servicio .....	55
Describe las acciones programadas para un objetivo escalable .....	57
Programa acciones de escalado recurrentes .....	58
Desactiva el escalado programado .....	61
Eliminación de una acción programada .....	62
Políticas de escalado de seguimiento de destino .....	64
Cómo funciona el seguimiento de objetivos .....	65
Funcionamiento .....	66
Elección de métricas .....	67
Definición del valor de destino .....	68
Defina los periodos de recuperación .....	68
Consideraciones .....	70
Políticas de escalado múltiples .....	71
Comandos de uso frecuente .....	72
Recursos relacionados .....	72
Limitaciones .....	72
Creación de una política de escalado de seguimiento de destino .....	73
Paso 1: Registrar un objetivo escalable .....	73
Paso 2: Crear una política de escalado de seguimiento de destino .....	74
Paso 3: Describe las políticas de escalado y seguimiento de objetivos .....	77
Eliminación de una política de escalado de seguimiento de destino .....	78
Uso de la calculadora de métricas .....	79
Ejemplo: cola de tareas pendientes de Amazon SQS por tarea .....	79
Limitaciones .....	84
Políticas de escalado por pasos .....	85
Cómo funciona el escalado por pasos .....	86
Funcionamiento .....	87
Ajustes de pasos .....	87
Tipos de ajuste de escalado .....	90
Periodo de recuperación .....	91
Comandos de uso frecuente .....	92
Consideraciones .....	92
Recursos relacionados .....	47

Acceso a la consola .....	93
Creación de una política de escalado por pasos .....	93
Paso 1: Registrar un objetivo escalable .....	93
Paso 2: Cree una política de escalado escalonado .....	94
Paso 3: Cree una alarma que invoque una política de escalado .....	98
Descripción de políticas de escalado por pasos .....	99
Eliminación de una política de escalado por pasos .....	101
Tutorial: configuración del escalado automático para manejar una carga de trabajo pesada .....	102
Requisitos previos .....	102
Paso 1: Registrar un onjetivo escalable .....	103
Paso 2: Configuración de acciones programadas según sus requisitos .....	104
Paso 3: Agregar una política de escalado de seguimiento de destino .....	108
Paso 4: Sigüientes pasos .....	110
Paso 5: Eliminar .....	111
Suspensión del escalado .....	113
Actividades de escalado .....	113
Suspender y reanudar las actividades de escalado .....	115
Ver actividades de escalado suspendidas .....	117
Reanudar actividades de escalado .....	118
Actividades de escalado .....	120
Busque las actividades de escalado por objetivo escalable .....	120
Incluya actividades no escaladas .....	121
Códigos de motivo .....	123
Supervisión .....	126
Supervise usando CloudWatch .....	127
CloudWatch métricas para monitorear el uso de los recursos .....	128
Métricas predefinidas para políticas de escalado de seguimiento de destino .....	141
AWS CloudTrail .....	145
Información sobre Application Auto Scaling en CloudTrail .....	145
Introducción a las entradas del archivo de registro de Application Auto Scaling .....	146
.....	146
Recursos relacionados .....	147
Amazon EventBridge .....	148
Eventos de Auto Scaling de aplicaciones .....	148
Compatibilidad del etiquetado .....	153
Ejemplos de etiquetas .....	153

---

Etiquetas para seguridad .....	154
Control del acceso a las etiquetas .....	155
Seguridad .....	157
Protección de datos .....	158
Identity and Access Management .....	159
Control de acceso .....	159
Cómo funciona Application Auto Scaling con IAM .....	159
AWS políticas gestionadas .....	165
Roles vinculados al servicio .....	176
Ejemplos de políticas basadas en identidades .....	182
Resolución de problemas .....	194
Validación de permisos .....	195
AWS PrivateLink .....	197
Cree un VPC punto final de interfaz .....	198
Cree una política de puntos VPC finales .....	198
Resiliencia .....	199
Seguridad de la infraestructura .....	199
Validación de conformidad .....	200
Cuotas .....	202
Historial de documentos .....	204
.....	ccxvi

# ¿Qué es Auto Scaling de aplicaciones?

Application Auto Scaling es un servicio web para desarrolladores y administradores de sistemas que necesitan una solución para escalar automáticamente sus recursos escalables para AWS servicios individuales más allá de AmazonEC2. Con Application Auto Scaling, puede configurar el escalado automático para los siguientes recursos:

- AppStream Flotas 2.0
- Réplicas de Aurora
- Puntos de conexión de reconocedor de identidades y clasificación de documentos de Amazon Comprehend
- Tablas de DynamoDB e índices secundarios globales
- ECSServicios de Amazon
- ElastiCache (RedisOSS) clústeres (grupos de replicación)
- EMRClústeres de Amazon
- Tablas de Amazon Keyspaces (for Apache Cassandra)
- Disponibilidad aprovisionada con la función Lambda
- Amazon Managed Streaming for Apache Kafka (MSK) broker Storage
- Clústeres de Amazon Neptune
- SageMaker variantes de punto final
- SageMaker componentes de inferencia
- SageMaker simultaneidad aprovisionada sin servidor
- Solicitudes de flota de spot
- Pool de Amazon WorkSpaces
- Los recursos personalizados proporcionados por sus propias aplicaciones o servicios. Para obtener más información, consulta el [GitHubrepositorio](#).

Para ver la disponibilidad regional de cualquiera de los AWS servicios enumerados anteriormente, consulte la tabla de [regiones Tabla](#) de .

Para obtener información sobre cómo escalar su flota de EC2 instancias de Amazon mediante grupos de Auto Scaling, consulte la [Guía del usuario de Amazon EC2 Auto Scaling](#).

# Características de Auto Scaling de aplicaciones

Auto Scaling de aplicaciones lo permite escalar automáticamente sus recursos escalables en función de las condiciones que defina.

- Escalado de seguimiento de objetivos: escale un recurso en función del valor objetivo de una CloudWatch métrica específica.
- Escalado por pasos: escala un recurso en función de un conjunto de ajustes de escalado que varían según el tamaño de la vulneración de la alarma.
- Escalado programado: escala un recurso solo una vez o según un programa periódico.

## Trabajar con Application Auto Scaling

Puede configurar el escalado con las siguientes interfaces según el recurso que esté escalando:

- AWS Management Console: proporciona una interfaz web que puede usar para configurar el escalado. Si se ha registrado para obtener una AWS cuenta, acceda a Application Auto Scaling iniciando sesión en AWS Management Console. A continuación, abra la consola del servicio para alguno de los recursos enumerados en la introducción. Asegúrese de abrir la consola en el Región de AWS mismo lugar que el recurso con el que desea trabajar.

### Note

El acceso de consola no está disponible para todos los recursos. Para obtener más información, consulte [Servicios de AWS que puede usar con Application Auto Scaling](#).

- AWS Command Line Interface (AWS CLI): proporciona comandos para un amplio conjunto de Servicios de AWS sistemas y es compatible con Windows, macOS y Linux. Para empezar, consulte [AWS Command Line Interface](#). Para obtener una lista de comandos, consulte [application-autoscaling en la AWS CLI Referencia](#) de comandos.
- AWS Tools for Windows PowerShell— Proporciona comandos para un amplio conjunto de AWS productos para quienes escriben en el entorno. PowerShell Para empezar, consulte la [AWS Tools for Windows PowerShell Guía del usuario de](#) . Para obtener más información, consulte la [Referencia de cmdlet de AWS Tools for PowerShell](#).
- AWS SDKs— Proporciona API operaciones específicas del idioma y se ocupa de muchos de los detalles de la conexión, como el cálculo de las firmas, la gestión de los reintentos de solicitudes

y la gestión de los errores. Para obtener más información, consulte [Herramientas sobre las que basarse](#). AWS

- HTTPSAPI— Proporciona API acciones de bajo nivel a las que se puede recurrir mediante HTTPS solicitudes. Para obtener más información, consulte la [APIReferencia de Application Auto Scaling](#).
- AWS CloudFormation— Admite la configuración del escalado mediante una CloudFormation plantilla. Para obtener más información, consulte [Configure los recursos de Application Auto Scaling mediante AWS CloudFormation](#).

Para conectarse mediante programación a un dispositivo Servicio de AWS, se utiliza un punto final. Para obtener información sobre los puntos de enlace de las llamadas a Application Auto Scaling, consulte los [puntos de enlace y las cuotas de Application Auto Scaling](#) en los Referencia general de AWS

## Conceptos de Application Auto Scaling

En este tema se explican los conceptos clave que le ayudarán a aprender acerca del Auto Scaling de aplicaciones y a empezar a utilizarlo.

### Destinos escalables

Entidad que se crea para especificar el recurso que desea escalar. Cada destino escalable se identifica de forma única mediante un espacio de nombres de servicio, un ID de recurso y una dimensión escalable, que representa alguna dimensión de capacidad del servicio subyacente. Por ejemplo, un Servicio ECS de Amazon admite el escalado automático de su recuento de tareas, una tabla de DynamoDB admite el escalado automático de la capacidad de lectura y escritura de la tabla y sus índices secundarios globales, y un clúster de Aurora admite el escalado de su recuento de réplicas.

#### Tip

Cada objetivo escalable también tiene una capacidad mínima y máxima. Las políticas de escalado nunca irán más alto o más bajo que el rango mínimo máximo. Puede realizar out-of-band cambios directamente en el recurso subyacente que estén fuera de este rango, algo que Application Auto Scaling desconoce. Sin embargo, cada vez que se invoca una política de escalado o se llama a la API `RegisterScalableTarget`, Auto Scaling de aplicaciones recupera la capacidad actual y la compara con la capacidad

mínima y máxima. Si está fuera del rango mínimo máximo, la capacidad se actualiza para cumplir con el mínimo y máximo establecido.

## Reducción horizontal

Cuando Auto Scaling de aplicaciones disminuye automáticamente la capacidad de un destino escalable, el destino escalable se reduce horizontalmente. Cuando se establecen políticas de escallAMento, no pueden reducir horizontalmente el objetivo escalable por debajo de su capacidad mínima.

## Escalado ascendente

Cuando Auto Scaling de aplicaciones aumenta automáticamente la capacidad de un destino escalable, el destino escalable escala horizontalmente. Cuando se establecen políticas de escallAMento, no pueden escalar horizontalmente el objetivo escalable por encima de su capacidad máxima.

## Política de escalado

Una política de escalado indica a Application Auto Scaling que realice un seguimiento de una CloudWatch métrica específica. A continuación, determina la acción de escala que se debe realizar cuando la métrica es superior o inferior a un determinado valor de umbral. Por ejemplo, es posible que desee escalar horizontalmente si el uso de la CPU en el clúster comienza a aumentar y reducir horizontalmente cuando vuelva a caer.

El servicio de destino publica las métricas que se utilizan para el escalado automático, pero también puedes publicar tu propia métrica CloudWatch y utilizarla después con una política de escalado.

Un periodo de recuperación entre actividades de escalado permite que el recurso se estabilice antes de que comience otra actividad de escalado. Auto Scaling de aplicaciones continúa evaluando métricas durante el periodo de recuperación. Cuando finaliza el periodo de recuperación, la política de escalado inicia otra actividad de escalado si es necesario. Mientras esté vigente un periodo de recuperación si se necesita una escala horizontal mayor en rol del valor de la métrica actual, la política de escalado se escala horizontalmente inmediatamente.

## Acción programada

Las acciones programadas escalan automáticamente los recursos en una fecha y hora específicas. Funcionan modificando la capacidad mínima y máxima de un destino escalable y,

por lo tanto, se pueden utilizar para reducir horizontalmente y escalar horizontalmente de una programación estableciendo la capacidad mínima alta o la capacidad máxima baja. Por ejemplo, puede usar acciones programadas para escalar una aplicación que no consume recursos los fines de semana reduciendo la capacidad el viernes y aumentando la capacidad el lunes siguiente.

También puede utilizar acciones programadas para optimizar los valores mínimo y máximo a lo largo del tiempo para adaptarse a situaciones en las que se espera un tráfico superior al normal, por ejemplo, campañas de marketing o fluctuaciones estacionales. Esto puede ayudarlo a mejorar el rendimiento en los momentos en que necesita escalar horizontalmente más alto para aumentar el uso y reducir los costos en momentos en que utiliza menos recursos.

## Más información

[Servicios de AWS que puede usar con Application Auto Scaling](#): Esta sección le presenta los servicios que puede escalar y le ayuda a configurar el escalado automático registrando un destino escalable. También describe cada uno de los roles vinculados al servicio de IAM que crea Auto Scaling de aplicaciones para acceder a los recursos del servicio de destino.

[Políticas de escalado de seguimiento de destino para Auto Scaling de aplicaciones](#): Una de las principales características de Auto Scaling de aplicaciones es las políticas de escalado de seguimiento de destino. Descubra cómo las políticas de seguimiento de destinos ajustan automáticamente la capacidad deseada para mantener la utilización en un nivel constante en rol de sus métricas y valores de destino configurados. Por ejemplo, puede configurar el seguimiento de destino para mantener el uso de la CPU para su flota de spot web en un 50 %. A continuación, Auto Scaling de aplicaciones lanza o finaliza instancias EC2 según sea necesario para mantener la utilización agregada de la CPU en todos los servidores en un 50 %.

# Servicios de AWS que puede usar con Application Auto Scaling

Application Auto Scaling se integra con otros AWS servicios para que pueda agregar capacidades de escalado para satisfacer la demanda de su aplicación. El escalado automático es una característica opcional del servicio que está desactivada de forma predeterminada en casi todos los casos.

En la siguiente tabla se enumeran los AWS servicios que puede usar con Application Auto Scaling, incluida información sobre los métodos compatibles para configurar el autoescalado. También puede utilizar Auto Scaling de aplicaciones con recursos personalizados.

- Acceso a la consola— Puede configurar un servicio AWS compatible para iniciar el escalado automático mediante la configuración de una política de escalado en la consola del servicio de destino.
- Acceso a la CLI— Puede configurar un servicio AWS compatible para iniciar el escalado automático mediante el AWS CLI.
- Acceso al SDK: puede configurar un AWS servicio compatible para iniciar el escalado automático mediante los AWS SDK.
- CloudFormation acceso: puede configurar un AWS servicio compatible para iniciar el escalado automático mediante una plantilla de AWS CloudFormation pila. Para obtener más información, consulte [Configure los recursos de Application Auto Scaling mediante AWS CloudFormation](#).

AWS servicio	Acceso a la consola <sup>1</sup>	Acceso a la CLI	Acceso al SDK	CloudFormation acceso
<a href="#">AppStream 2.0</a>	 Sí	 Sí	 Sí	 Sí
<a href="#">Aurora</a>	 Sí	 Sí	 Sí	 Sí

AWS servicio	Acceso a la consola <sup>1</sup>	Acceso a la CLI	Acceso al SDK	CloudFormation acceso
<a href="#">Amazon Comprehend</a>	 No	 Sí	 Sí	 Sí
<a href="#">Amazon DynamoDB</a>	 Sí	 Sí	 Sí	 Sí
<a href="#">Amazon ECS</a>	 Sí	 Sí	 Sí	 Sí
<a href="#">Amazon ElastiCache</a>	 Sí	 Sí	 Sí	 Sí
<a href="#">Amazon EMR</a>	 Sí	 Sí	 Sí	 Sí
<a href="#">Amazon Keyspaces</a>	 Sí	 Sí	 Sí	 Sí
<a href="#">Lambda</a>	 No	 Sí	 Sí	 Sí
<a href="#">Amazon MSK</a>	 Sí	 Sí	 Sí	 Sí

AWS servicio	Acceso a la consola <sup>1</sup>	Acceso a la CLI	Acceso al SDK	CloudFormation acceso
<a href="#">Amazon Neptune</a>	 No	 Sí	 Sí	 Sí
<a href="#">SageMaker</a>	 Sí	 Sí	 Sí	 Sí
<a href="#">Flota de spot</a>	 Sí	 Sí	 Sí	 Sí
<a href="#">WorkSpaces</a>	 Sí	 Sí	 Sí	 Sí
<a href="#">Recursos personales</a>	 No	 Sí	 Sí	 Sí

<sup>1</sup> Acceso a la consola para configurar las políticas de escalado. La mayoría de los servicios no admiten la configuración del escalado programado desde la consola. Actualmente, solo Amazon AppStream 2.0 y Spot Fleet ofrecen acceso a la consola para el escalado programado. ElastiCache

## Amazon AppStream 2.0 y Application Auto Scaling

Puede escalar flotas AppStream 2.0 mediante políticas de escalado de seguimiento de objetivos, políticas de escalado escalonado y escalado programado.

Utilice la siguiente información para ayudarle a integrar AppStream 2.0 con Application Auto Scaling.

## Función vinculada al servicio creada para la versión 2.0 AppStream

El siguiente [rol vinculado al servicio](#) se crea automáticamente en usted Cuenta de AWS al registrar los recursos AppStream 2.0 como objetivos escalables con Application Auto Scaling. Este rol permite que Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- `AWSServiceRoleForApplicationAutoScaling_AppStreamFleet`

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso a la siguiente entidad de seguridad de servicio:

- `appstream.application-autoscaling.amazonaws.com`

## Registro de flotas AppStream 2.0 como objetivos escalables con Application Auto Scaling

Application Auto Scaling requiere un objetivo escalable antes de poder crear políticas de escalado o acciones programadas para una flota AppStream 2.0. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Si configura el escalado automático mediante la consola AppStream 2.0, la AppStream versión 2.0 registrará automáticamente un objetivo escalable.

Si desea configurar el escalado automático mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Llame al [register-scalable-target](#) comando para una flota AppStream 2.0. En el siguiente ejemplo se registra la capacidad deseada de una flota llamada `sample-fleet`, con una capacidad mínima de una instancia de flota y una capacidad máxima de cinco instancias de flota.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace appstream \  
  --scalable-dimension appstream:fleet:DesiredCapacity \  
  --resource-id fleet/sample-fleet \  
  --min-capacity 1 \  
  --max-capacity 5
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione ResourceId, ScalableDimension, ServiceNamespace, MinCapacity y MaxCapacity como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre cómo escalar sus recursos AppStream 2.0 en la siguiente documentación:

[Fleet Auto Scaling for AppStream 2.0](#) en la Guía de administración de Amazon AppStream 2.0

## Auto Scaling de aplicaciones y Amazon Aurora

Puede escalar los clústeres de base de datos de Aurora mediante políticas de escalado de seguimiento de destino, políticas de escalado por pasos y escalado programado.

Utilice la siguiente información para ayudarle a integrar Aurora con Auto Scaling de aplicaciones.

### Rol vinculado al servicio creado para Aurora

El siguiente [rol vinculado al servicio](#) se crea automáticamente en usted Cuenta de AWS al registrar los recursos de Aurora como destinos escalables con Application Auto Scaling. Este rol permite que Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- `AWSServiceRoleForApplicationAutoScaling_RDSCluster`

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso a la siguiente entidad de seguridad de servicio:

- `rds.application-autoscaling.amazonaws.com`

## Registro de clústeres de Aurora DB como destinos escalables con Auto Scaling de aplicaciones

Auto Scaling de aplicaciones requiere un destino escalable antes de que pueda crear políticas de escalado o acciones programadas para un clúster de Aurora. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Si configura el escalado automático mediante la consola de Aurora, Aurora registra automáticamente un destino escalable para usted.

Si desea configurar el escalado automático mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Llame al comando [register-scalable-target](#) para un clúster de Aurora. En el siguiente ejemplo se registra el recuento de réplicas de Aurora en un clúster denominado `my-db-cluster`, con una capacidad mínima de una réplica de Aurora y una capacidad máxima de ocho réplicas de Aurora.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace rds \  
  --scalable-dimension rds:cluster:ReadReplicaCount \  
  --resource-id cluster:my-db-cluster \  
  --min-capacity 1 \  
  --max-capacity 8
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione ResourceId, ScalableDimension, ServiceNamespace, MinCapacity y MaxCapacity como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre cómo escalar los recursos de Aurora en la siguiente documentación:

[Uso de Auto Scaling de Amazon Aurora con réplicas de Aurora](#); en la guía del usuario de Amazon RDS

## Auto Scaling de aplicaciones y Amazon Comprehend

Puede escalar la clasificación de documentos de Amazon Comprehend y los extremos del reconocedor de entidades mediante políticas de escalado de seguimiento de destino y escalado programado.

Utilice la siguiente información para ayudarle a integrar Amazon Comprehend con Auto Scaling de aplicaciones.

### Se ha creado un rol vinculado al servicio para Amazon Comprehend

El siguiente [rol vinculado al servicio](#) se crea automáticamente en su Cuenta de AWS al registrar los recursos de Amazon Comprehend como destinos escalables con Application Auto Scaling. Este rol permite que Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- AWSServiceRoleForApplicationAutoScaling\_ComprehendEndpoint

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso la siguiente entidad de seguridad de servicio:

- `comprehend.application-autoscaling.amazonaws.com`

## Registro de recursos de Amazon Comprehend como destinos escalables con Auto Scaling de aplicaciones

Auto Scaling de aplicaciones requiere un destino escalable antes de poder crear políticas de escalado o acciones programadas para una clasificación de documentos de Amazon Comprehend o punto de enlace del reconocedor de entidades. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Para configurar el autoescalado mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Llamada al comando [register-scalable-target](#) para un punto de enlace de clasificación de documentos. En el ejemplo siguiente se registra el número deseado de unidades de inferencia que utilizará el modelo para un punto de enlace del clasificador de documentos utilizando el ARN del extremo, con una capacidad mínima de una unidad de inferencia y una capacidad máxima de tres unidades de inferencia.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace comprehend \  
  --scalable-dimension comprehend:document-classifier-endpoint:DesiredInferenceUnits \  
  \  
  --resource-id arn:aws:comprehend:us-west-2:123456789012:document-classifier-   
  endpoint/EXAMPLE \  
  --min-capacity 1 \  
  --max-capacity 3
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Llamada a un comando [register-scalable-target](#) para un punto de enlace del reconocedor de entidades. En el ejemplo siguiente se registra el número deseado de unidades de inferencia que el modelo utilizará para un reconocedor de entidades utilizando el ARN del extremo, con una capacidad mínima de una unidad de inferencia y una capacidad máxima de tres unidades de inferencia.

```
aws application-autoscaling register-scalable-target \
  --service-namespace comprehend \
  --scalable-dimension comprehend:entity-recognizer-endpoint:DesiredInferenceUnits \
  --resource-id arn:aws:comprehend:us-west-2:123456789012:entity-recognizer-
endpoint/EXAMPLE \
  --min-capacity 1 \
  --max-capacity 3
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione `ResourceId`, `ScalableDimension`, `ServiceNamespace`, `MinCapacity` y `MaxCapacity` como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre cómo escalar sus recursos de Amazon Comprehend en la siguiente documentación:

[Escalado automático con puntos de enlace](#) en la Guía para desarrolladores de Amazon Comprehend

# Auto Scaling de aplicaciones y Amazon DynamoDB

Puede escalar tablas de DynamoDB e índices secundarios globales mediante políticas de escalado de seguimiento de destino y escalado programado.

Utilice la siguiente información para ayudarle a integrar DynamoDB con Auto Scaling de aplicaciones.

## Rol vinculado al servicio creado para DynamoDB

El siguiente [rol vinculado al servicio](#) se crea automáticamente en usted Cuenta de AWS al registrar los recursos de DynamoDB como destinos escalables con Application Auto Scaling. Este rol permite que Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- `AWSServiceRoleForApplicationAutoScaling_DynamoDBTable`

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso a la siguiente entidad de seguridad de servicio:

- `dynamodb.application-autoscaling.amazonaws.com`

## Registro de recursos de DynamoDB como destinos escalables con Auto Scaling de aplicaciones

El Auto Scaling de aplicaciones requiere un destino escalable antes de que pueda crear políticas de escalado o acciones programadas para una tabla de DynamoDB o un índice secundario global. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Si configura el escalado automático mediante la consola de DynamoDB, DynamoDB registra automáticamente un destino escalable.

Si desea configurar el escalado automático mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Llame al [register-scalable-target](#) comando para conocer la capacidad de escritura de una tabla. En el siguiente ejemplo, se registra la capacidad de escritura aprovisionada de una tabla denominada `my-table`, con una capacidad mínima de cinco unidades de capacidad de escritura y una capacidad máxima de 10 unidades de capacidad de escritura.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace dynamodb \  
  --scalable-dimension dynamodb:table:WriteCapacityUnits \  
  --resource-id table/my-table \  
  --min-capacity 5 \  
  --max-capacity 10
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

Llame al [register-scalable-target](#) comando para conocer la capacidad de lectura de una tabla. El siguiente ejemplo registra la capacidad de lectura aprovisionada de una tabla denominada `my-table`, con una capacidad mínima de cinco unidades de capacidad de lectura y una capacidad máxima de diez unidades de lectura.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace dynamodb \  
  --scalable-dimension dynamodb:table:ReadCapacityUnits \  
  --resource-id table/my-table \  
  --min-capacity 5 \  
  --max-capacity 10
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
```

```
"ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Llame al [register-scalable-target](#) comando para conocer la capacidad de escritura de un índice secundario global. En el siguiente ejemplo, se registra la capacidad de escritura aprovisionada de una tabla denominada `my-table-index`, con una capacidad mínima de cinco unidades de capacidad de escritura y una capacidad máxima de 10 unidades de capacidad de escritura.

```
aws application-autoscaling register-scalable-target \
  --service-namespace dynamodb \
  --scalable-dimension dynamodb:index:WriteCapacityUnits \
  --resource-id table/my-table/index/my-table-index \
  --min-capacity 5 \
  --max-capacity 10
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

Ejecute el [register-scalable-target](#) comando para obtener la capacidad de lectura de un índice secundario global. En el siguiente ejemplo, se registra la capacidad de lectura aprovisionada de un índice secundario global denominado `my-table-index`, con una capacidad mínima de cinco unidades de capacidad de lectura y una capacidad máxima de diez unidades de capacidad de lectura.

```
aws application-autoscaling register-scalable-target \
  --service-namespace dynamodb \
  --scalable-dimension dynamodb:index:ReadCapacityUnits \
  --resource-id table/my-table/index/my-table-index \
  --min-capacity 5 \
  --max-capacity 10
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
```

```
"ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

- AWS SDK:

Llame a la [RegisterScalableTarget](#) operación y proporcione ResourceId, ScalableDimensionServiceNamespace, MinCapacity, y MaxCapacity como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, encontrará información adicional útil sobre el escalado de los recursos de DynamoDB en la siguiente documentación:

- [Administración de la capacidad de rendimiento con el Auto Scaling de DynamoDB](#) en la Guía para desarrolladores de Amazon DynamoDB
- [Evalúe la configuración de escalado automático de su tabla](#) en la Guía para desarrolladores de Amazon DynamoDB
- [Cómo configurar el AWS CloudFormation escalado automático para tablas e índices de DynamoDB](#) en el blog AWS

## Amazon ECS y Auto Scaling de aplicaciones

Puede escalar servicios ECS con las políticas de escalado de seguimiento de destino, las políticas de escalado por pasos y el escalado programado.

Utilice la siguiente información para ayudarle a integrar Amazon ECS con Auto Scaling de aplicaciones.

### Se ha creado un rol vinculado al servicio para Amazon ECS

El siguiente [rol vinculado al servicio](#) se crea automáticamente en su Cuenta de AWS al registrar los recursos de Amazon ECS como destinos escalables con Application Auto Scaling. Este rol permite que Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- AWSServiceRoleForApplicationAutoScaling\_ECSService

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso a la siguiente entidad de seguridad de servicio:

- `ecs.application-autoscaling.amazonaws.com`

## Registro de servicios ECS como destinos escalables con Auto Scaling de aplicaciones

Auto Scaling de aplicaciones requiere un destino escalable antes de poder crear políticas de escalado o acciones programadas para un Servicio ECS de Amazon. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Si configura el escalado automático mediante la consola de Amazon ECS, Amazon ECS registra automáticamente un destino escalable para usted.

Si desea configurar el escalado automático mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Llame al comando [register-scalable-target](#) para un Servicio ECS de Amazon. El siguiente ejemplo registra un destino escalable para un servicio llamado `sample-app-service`, que se ejecuta en el clúster `default`, con un recuento mínimo de tareas de una tarea y un recuento máximo de tareas de 10 tareas.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace ecs \  
  --scalable-dimension ecs:service:DesiredCount \  
  --resource-id service/default/sample-app-service \  
  --min-capacity 1 \  
  --max-capacity 10
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione ResourceId, ScalableDimension, ServiceNamespace, MinCapacity y MaxCapacity como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre cómo escalar los recursos de Amazon ECS en la siguiente documentación:

- El [escalado automático de servicios](#) en la guía para desarrolladores de Amazon Elastic Container Service
- [Configuración del escalado automático de servicios](#) en la Guía de prácticas recomendadas de Amazon Elastic Container Service

### Note

Para obtener instrucciones sobre cómo suspender los procesos de escalado horizontal mientras las implementaciones de Amazon ECS están en curso, consulte la siguiente documentación:

[Implementaciones y escalado automáticos de servicios](#) en la Guía para desarrolladores de Amazon Elastic Container Service

## ElastiCache (RedisOSS) y Application Auto Scaling

Puede escalar los grupos de replicación ElastiCache (RedisOSS) mediante el seguimiento de objetivos, las políticas de escalado y el escalado programado.

Utilice la siguiente información para ayudarle a integrarse ElastiCache con Application Auto Scaling.

## Rol vinculado a un servicio creado para ElastiCache

El siguiente [rol vinculado al servicio](#) se crea automáticamente en usted Cuenta de AWS al registrar ElastiCache los recursos como destinos escalables con Application Auto Scaling. Este rol permite que Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- `AWSServiceRoleForApplicationAutoScaling_ElastiCacheRG`

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso a la siguiente entidad de seguridad de servicio:

- `elasticache.application-autoscaling.amazonaws.com`

## Registro de grupos de replicación ElastiCache (RedisOSS) como destinos escalables con Application Auto Scaling

Application Auto Scaling requiere un objetivo escalable antes de poder crear políticas de escalado o acciones programadas para un grupo de ElastiCache replicación. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Si configura el escalado automático mediante la ElastiCache consola, entonces registra ElastiCache automáticamente un objetivo escalable para usted.

Si desea configurar el autoescalado mediante una AWS CLI o una de las opciones AWS SDKs, puede utilizar las siguientes opciones:

- AWS CLI:

Llame al [register-scalable-target](#) comando de un grupo de ElastiCache replicación. En el siguiente ejemplo se registra el número deseado de grupos de nodo para un grupo de replicación denominado `mycluster`, con una capacidad mínima de uno y una capacidad máxima de cinco.

```
aws application-autoscaling register-scalable-target \
  --service-namespace elasticache \
  --scalable-dimension elasticache:replication-group:NodeGroups \
  --resource-id replication-group/mycluster \
  --min-capacity 1 \
  --max-capacity 5
```

Si se ejecuta correctamente, este comando devuelve el ARN del objetivo escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

En el ejemplo siguiente se registra el número deseado de réplicas por grupo de nodos para un grupo de replicación denominado *mycluster*, con una capacidad mínima de 1 y una capacidad máxima de 5.

```
aws application-autoscaling register-scalable-target \
  --service-namespace elasticache \
  --scalable-dimension elasticache:replication-group:Replicas \
  --resource-id replication-group/mycluster \
  --min-capacity 1 \
  --max-capacity 5
```

Si se ejecuta correctamente, este comando devuelve el ARN del objetivo escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

- AWS SDK:

Llame a la [RegisterScalableTarget](#) operación y proporcione `ResourceId`, `ScalableDimensionServiceNamespace`, `MinCapacity`, y `MaxCapacity` como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre cómo escalar sus ElastiCache recursos en la siguiente documentación:

[Clústeres de Auto Scaling ElastiCache \(RedisOSS\)](#) en la Guía del usuario de Amazon ElastiCache (RedisOSS)

## Amazon Keyspaces (for Apache Cassandra) y Auto Scaling de aplicaciones

Puede escalar las tablas de Amazon Keyspaces mediante políticas de escalado de seguimiento de destino y escalado programado.

Utilice la siguiente información para ayudarle a integrar Amazon Keyspaces con Auto Scaling de aplicaciones.

### Se ha creado un rol vinculado al servicio para Amazon Keyspaces

El siguiente [rol vinculado al servicio](#) se crea automáticamente en usted Cuenta de AWS al registrar los recursos de Amazon Keyspaces como destinos escalables con Application Auto Scaling. Este rol permite que Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- `AWSServiceRoleForApplicationAutoScaling_CassandraTable`

### Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso a la siguiente entidad de seguridad de servicio:

- `cassandra.application-autoscaling.amazonaws.com`

## Registro de tablas de Amazon Keyspaces como destinos escalables con Auto Scaling de aplicaciones

Auto Scaling de aplicaciones requiere un destino escalable antes de poder crear políticas de escalado o acciones programadas para una tabla de Amazon Keyspaces. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Si configura el escalado automático mediante la consola de Amazon Keyspaces, Amazon Keyspaces registra automáticamente un destino escalable.

Si desea configurar el escalado automático mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Llame al comando [register-scalable-target](#) para una tabla de Amazon Keyspaces. En el siguiente ejemplo, se registra la capacidad de escritura aprovisionada de una tabla denominada `mytable`, con una capacidad mínima de cinco unidades de capacidad de escritura y una capacidad máxima de 10 unidades de capacidad de escritura.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace cassandra \  
  --scalable-dimension cassandra:table:WriteCapacityUnits \  
  --resource-id keyspace/mykeyspace/table/mytable \  
  --min-capacity 5 \  
  --max-capacity 10
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

En el siguiente ejemplo, se registra la capacidad de lectura aprovisionada de una tabla denominada `mytable`, con una capacidad mínima de cinco unidades de capacidad de lectura y una capacidad máxima de 10 unidades de capacidad de lectura.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace cassandra \  
  --scalable-dimension cassandra:table:ReadCapacityUnits \  
  --resource-id keyspace/mykeyspace/table/mytable \  
  --min-capacity 5 \  
  --max-capacity 10
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione ResourceId, ScalableDimension, ServiceNamespace, MinCapacity y MaxCapacity como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre cómo escalar sus recursos de Amazon Keyspaces en la siguiente documentación:

[Administración de la capacidad de rendimiento con el escalado automático de Amazon Keyspaces](#) en la Guía para desarrolladores de Amazon Keyspaces (para Apache Cassandra)

## AWS Lambda y Application Auto Scaling

Puede escalar la simultaneidad AWS Lambda aprovisionada mediante el seguimiento de objetivos, las políticas de escalado y el escalado programado.

Utilice la siguiente información para ayudarle a integrar Lambda con Auto Scaling de aplicaciones.

## Rol vinculado al servicio creado para Lambda

El siguiente [rol vinculado al servicio](#) se crea automáticamente en usted Cuenta de AWS al registrar los recursos de Lambda como destinos escalables con Application Auto Scaling. Este rol permite que

Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- `AWSServiceRoleForApplicationAutoScaling_LambdaConcurrency`

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso al siguiente maestro de servicio:

- `lambda.application-autoscaling.amazonaws.com`

## Registrar las funciones de Lambda como destinos escalables con Auto Scaling de aplicaciones

Auto Scaling de aplicaciones requiere un destino escalable antes de que pueda crear políticas de escalado o acciones programadas para una función Lambda. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Para configurar el autoescalado mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Llame al comando [register-scalable-target](#) para una función Lambda. En el ejemplo siguiente se registra la concurrencia aprovisionada para un alias denominado BLUE para una función denominada `my-function`, con una capacidad mínima de 0 y una capacidad máxima de 100.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace lambda \  
  --scalable-dimension lambda:function:ProvisionedConcurrency \  
  --resource-id function:my-function:BLUE \  
  --min-capacity 0 \  
  --max-capacity 100
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione ResourceId, ScalableDimension, ServiceNamespace, MinCapacity y MaxCapacity como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre el escalado de las funciones de Lambda en la siguiente documentación:

- [Cómo configurar la simultaneidad aprovisionada en la Guía](#) para desarrolladores AWS Lambda
- [Programar la simultaneidad aprovisionada por Lambda para los picos de uso recurrentes](#) en el blog AWS

## Amazon Managed Streaming for Apache Kafka (MSK) y Auto Scaling de aplicaciones

Puede escalar horizontalmente el almacenamiento de clústeres de Amazon MSK mediante políticas de escalado de seguimiento de destino. La reducción horizontal por la política de seguimiento de destino está desactivada.

Utilice la siguiente información para ayudarle a integrar Amazon MSK con Auto Scaling de aplicaciones.

### Se ha creado un rol vinculado al servicio para Amazon MSK

El siguiente [rol vinculado al servicio](#) se crea automáticamente en usted Cuenta de AWS al registrar los recursos de Amazon MSK como destinos escalables con Application Auto Scaling. Este rol permite que Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- `AWSServiceRoleForApplicationAutoScaling_KafkaCluster`

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso a la siguiente entidad de seguridad de servicio:

- `kafka.application-autoscaling.amazonaws.com`

## Registro del almacenamiento en clúster de Amazon MSK como destinos escalables con Auto Scaling de aplicaciones

Auto Scaling de aplicaciones requiere un destino escalable antes de que pueda crear una política de escalado para el tamaño del volumen de almacenamiento por agente de un clúster de Amazon MSK. Un destino escalable es un recurso que Auto Scaling de aplicaciones puede escalar. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Si configura el escalado automático mediante la consola de Amazon MSK, Amazon MSK registra automáticamente un destino escalable para usted.

Si desea configurar el escalado automático mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Llamada al comando [register-scalable-target](#) para un clúster de Amazon MSK. En el ejemplo siguiente se registra el tamaño del volumen de almacenamiento por agente de un clúster de Amazon MSK, con una capacidad mínima de 100 GiB y una capacidad máxima de 800 GiB.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace kafka \  
  --scalable-dimension kafka:broker-storage:VolumeSize \  
  --resource-id arn:aws:kafka:us-east-1:123456789012:cluster/demo-  
cluster-1/6357e0b2-0e6a-4b86-a0b4-70df934c2e31-5 \  
  --min-capacity 100 \  

```

```
--max-capacity 800
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione `ResourceId`, `ScalableDimension`, `ServiceNamespace`, `MinCapacity` y `MaxCapacity` como parámetros.

 Note

Cuando un clúster de Amazon MSK es el destino escalable, reducir horizontalmente se desactiva y no se puede habilitar.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre cómo escalar sus recursos de Amazon MSK en la siguiente documentación:

El [escalado automático](#) en la guía para desarrolladores de Amazon Managed Streaming for Apache Kafka

## Amazon Neptune y Auto Scaling de aplicaciones

Puede escalar clústeres de Neptune mediante políticas de escalado de seguimiento de destino y escalado programado.

Utilice la siguiente información como ayuda para integrar Neptune con Auto Scaling de aplicaciones.

## Rol vinculado a servicio creado para Neptune

El siguiente [rol vinculado al servicio](#) se crea automáticamente en usted Cuenta de AWS al registrar los recursos de Neptune como destinos escalables con Application Auto Scaling. Este rol permite que

Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- `AWSServiceRoleForApplicationAutoScaling_NeptuneCluster`

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso a la siguiente entidad de seguridad de servicio:

- `neptune.application-autoscaling.amazonaws.com`

## Registro de clústeres de Neptune como destinos escalables con Auto Scaling de aplicaciones

Auto Scaling de aplicaciones requiere un destino escalable para que se puedan crear políticas de escalado o acciones programadas para un clúster de Neptune. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Para configurar el autoescalado mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Ejecute el [register-scalable-target](#) comando para un cúmulo de Neptuno. En el siguiente ejemplo, se registra la capacidad deseada de un clúster denominado `mycluster`, con un valor de capacidad mínima de 1 y un valor de capacidad máxima de 8.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace neptune \  
  --scalable-dimension neptune:cluster:ReadReplicaCount \  
  --resource-id cluster:mycluster \  
  --min-capacity 1 \  
  --max-capacity 8
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione ResourceId, ScalableDimension, ServiceNamespace, MinCapacity y MaxCapacity como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre cómo escalar sus recursos de Neptune en la siguiente documentación:

[Escalado automático del número de réplicas de un clúster de bases de datos de Amazon Neptune](#) en la Guía del usuario de Neptune

## Amazon SageMaker y Application Auto Scaling

Puede escalar las variantes de los puntos SageMaker finales, la simultaneidad provisionada para los puntos finales sin servidor y los componentes de inferencia mediante políticas de escalado de seguimiento de objetivos, políticas de escalado escalonado y escalado programado.

Utilice la siguiente información para ayudarle a integrarse SageMaker con Application Auto Scaling.

### Rol vinculado a un servicio creado para SageMaker

El siguiente [rol vinculado al servicio](#) se crea automáticamente en usted Cuenta de AWS al registrar SageMaker los recursos como destinos escalables con Application Auto Scaling. Este rol permite que Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- AWSServiceRoleForApplicationAutoScaling\_SageMakerEndpoint

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso a la siguiente entidad de seguridad de servicio:

- `sagemaker.application-autoscaling.amazonaws.com`

## Registro de SageMaker variantes de puntos finales como objetivos escalables con Application Auto Scaling

Application Auto Scaling requiere un objetivo escalable antes de poder crear políticas de escalado o acciones programadas para un SageMaker modelo (variante). Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Si configura el escalado automático mediante la SageMaker consola, entonces registra SageMaker automáticamente un objetivo escalable para usted.

Si desea configurar el escalado automático mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Ejecute el [register-scalable-target](#) comando para obtener una variante del producto. En el ejemplo siguiente se registra el recuento de instancias deseado para una variante de producto denominada `my-variant`, que se ejecuta en el punto de enlace `my-endpoint`, con una capacidad mínima de una instancia y una capacidad máxima de ocho instancias.

```
aws application-autoscaling register-scalable-target \
  --service-namespace sagemaker \
  --scalable-dimension sagemaker:variant:DesiredInstanceCount \
  --resource-id endpoint/my-endpoint/variant/my-variant \
  --min-capacity 1 \
  --max-capacity 8
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione `ResourceId`, `ScalableDimension`, `ServiceNamespace`, `MinCapacity` y `MaxCapacity` como parámetros.

## Registro de la simultaneidad de puntos de conexión sin servidor como destinos escalables con Application Auto Scaling

Application Auto Scaling también requiere un destino escalable antes de poder crear políticas de escalado o acciones programadas para la simultaneidad de puntos de conexión sin servidor.

Si configura el escalado automático mediante la SageMaker consola, entonces registra SageMaker automáticamente un objetivo escalable para usted.

De lo contrario, utilice uno de los siguientes métodos para registrar el destino escalable:

- AWS CLI:

Ejecuta el [register-scalable-target](#) comando para obtener una variante del producto. En el ejemplo siguiente, se registra la simultaneidad aprovisionada para una variante de producto denominada `my-variant`, que se ejecuta en el punto de conexión `my-endpoint`, con una capacidad mínima de una instancia y una capacidad máxima de diez instancias.

```
aws application-autoscaling register-scalable-target \
  --service-namespace sagemaker \
  --scalable-dimension sagemaker:variant:DesiredProvisionedConcurrency \
  --resource-id endpoint/my-endpoint/variant/my-variant \
  --min-capacity 1 \
  --max-capacity 10
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
```

```
"ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione ResourceId, ScalableDimension, ServiceNamespace, MinCapacity y MaxCapacity como parámetros.

## Registro de clústeres de componentes de inferencia como destinos escalables con Application Auto Scaling

Application Auto Scaling también requiere un destino escalable para que se puedan crear políticas de escalado o acciones programadas para componentes de inferencia.

- AWS CLI:

Llame al [register-scalable-target](#) comando de un componente de inferencia. En el ejemplo siguiente se registra el recuento deseado para un componente de inferencia denominado `my-inference-component`, con una capacidad mínima de cero copias y una capacidad máxima de tres copias.

```
aws application-autoscaling register-scalable-target \
  --service-namespace sagemaker \
  --scalable-dimension sagemaker:inference-component:DesiredCopyCount \
  --resource-id inference-component/my-inference-component \
  --min-capacity 0 \
  --max-capacity 3
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione ResourceId, ScalableDimension, ServiceNamespace, MinCapacity y MaxCapacity como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre cómo escalar sus SageMaker recursos en la Guía para SageMaker desarrolladores de Amazon:

- [Escala automáticamente los SageMaker modelos de Amazon](#)
- [Escale automáticamente la simultaneidad aprovisionada para un punto final sin servidor](#)
- [Establezca políticas de autoescalado para despliegues de terminales multimodelo](#)
- [Escalar automáticamente un punto final asíncrono](#)

### Note

En 2023, SageMaker introdujo nuevas capacidades de inferencia basadas en puntos finales de inferencia en tiempo real. Se crea un SageMaker punto final con una configuración de punto final que define el tipo de instancia y el recuento inicial de instancias del punto final. A continuación, cree un componente de inferencia, que es un objeto de SageMaker alojamiento que puede utilizar para implementar un modelo en un punto final. Para obtener información sobre cómo escalar los componentes de inferencia, consulte [Amazon SageMaker agrega nuevas capacidades de inferencia para ayudar a reducir los costos y la latencia de implementación del modelo básico](#) y [reduce los costos de implementación del modelo en un 50% de media utilizando las últimas funciones de Amazon SageMaker](#) en el AWS blog.

## Auto Scaling de aplicaciones y flota de spot de Amazon EC2

Puede escalar las flotas de spot mediante las políticas de escalado de seguimiento de destino, las políticas de escalado por pasos y el escalado programado.

Utilice la siguiente información para ayudarle a integrar la flota de spot con Auto Scaling de aplicaciones.

### Rol vinculado al servicio creado para la flota de spot

El siguiente [rol vinculado al servicio](#) se crea automáticamente en usted Cuenta de AWS al registrar los recursos de Spot Fleet como objetivos escalables con Application Auto Scaling. Este rol permite

que Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- `AWSServiceRoleForApplicationAutoScaling_EC2SpotFleetRequest`

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso a la siguiente entidad de seguridad de servicio:

- `ec2.application-autoscaling.amazonaws.com`

## Registro de flotas de spot como destinos escalables con Auto Scaling de aplicaciones

El Auto Scaling de aplicaciones requiere un destino escalable antes de que pueda crear políticas de escalado o acciones programadas para una flota de spot. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Si configura el escalado automático mediante la consola flota de spot, a continuación, flota de spot registra automáticamente un destino escalable para usted.

Si desea configurar el escalado automático mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Llame al comando [register-scalable-target](#) para una flota de spot. En el ejemplo siguiente se registra la capacidad de destino de una flota de suspot utilizando su ID de solicitud, con una capacidad mínima de dos instancias y una capacidad máxima de 10 instancias.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace ec2 \  
  --scalable-dimension ec2:spot-fleet-request:TargetCapacity \  
  --target-id <ID de solicitud>
```

```
--resource-id spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \  
--min-capacity 2 \  
--max-capacity 10
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione ResourceId, ScalableDimension, ServiceNamespace, MinCapacity y MaxCapacity como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre cómo escalar su flota de spot en la siguiente documentación:

[Escalado automático de flota de spot](#) en la Guía del usuario de Amazon EC2

## Amazon WorkSpaces y Application Auto Scaling

Puede ampliar un grupo WorkSpaces utilizando políticas de escalado de seguimiento de objetivos, políticas de escalado escalonado y escalado programado.

Utilice la siguiente información para ayudarle a integrarse WorkSpaces con Application Auto Scaling.

### Rol vinculado a un servicio creado para WorkSpaces

Application Auto Scaling crea automáticamente el rol vinculado al servicio nombrado `AWSServiceRoleForApplicationAutoScaling_WorkSpacesPool` en su nombre Cuenta de AWS cuando registra WorkSpaces los recursos como objetivos escalables con Application Auto Scaling. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

Este rol vinculado al servicio usa la política administrada.

`AWSApplicationAutoscalingWorkSpacesPoolPolicy` Esta política otorga a Application Auto Scaling

permisos para llamar a Amazon WorkSpaces en tu nombre. Para obtener más información, consulte [AWSApplicationAutoscalingWorkSpacesPoolPolicy](#) la Referencia de políticas AWS gestionadas.

## Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

La función vinculada al servicio confía en que el siguiente director de servicio asuma la función:

- `workspaces.application-autoscaling.amazonaws.com`

## Registro de WorkSpaces grupos como objetivos escalables con Application Auto Scaling

Application Auto Scaling requiere un objetivo escalable antes de poder crear políticas de escalado o acciones programadas para él WorkSpaces. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Si configura el escalado automático mediante la WorkSpaces consola, entonces registra WorkSpaces automáticamente un objetivo escalable para usted.

Si desea configurar el escalado automático mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Ejecute el [register-scalable-target](#) comando para obtener un conjunto de WorkSpaces. El siguiente ejemplo registra la capacidad objetivo de un grupo WorkSpaces utilizando su identificador de solicitud, con una capacidad mínima de dos escritorios virtuales y una capacidad máxima de diez escritorios virtuales.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace workspaces \  
  --resource-id workspacespool/wspool-abcdef012 \  
  --scalable-dimension workspaces:workspacespool:DesiredUserSessions \  
  --min-capacity 2 \  
  --max-capacity 10
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

- AWS SDK:

Llame a la [RegisterScalableTarget](#) operación y proporcione ResourceId, ScalableDimensionServiceNamespace, MinCapacity, y MaxCapacity como parámetros.

## Recursos personalizados y Auto Scaling de aplicaciones

Puede escalar recursos personalizados mediante las políticas de escalado de seguimiento de destino, las políticas de escalado por pasos y el escalado programado.

Utilice la siguiente información para ayudarle a integrar recursos personalizados con Auto Scaling de aplicaciones.

### Rol vinculado al servicio creado para recursos personalizados

El siguiente [rol vinculado al servicio](#) se crea automáticamente en usted Cuenta de AWS al registrar los recursos personalizados como destinos escalables con Application Auto Scaling. Este rol permite que Auto Scaling de aplicaciones realice operaciones compatibles dentro de su cuenta. Para obtener más información, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

- AWSServiceRoleForApplicationAutoScaling\_CustomResource

### Entidad de seguridad de servicio utilizada por el rol vinculado al servicio

El rol vinculado al servicio de la sección anterior solo puede ser asumido por la entidad de seguridad de servicio autorizada por las relaciones de confianza definidas para el rol. El rol vinculado al servicio utilizado por Auto Scaling de aplicaciones concede acceso a la siguiente entidad de seguridad de servicio:

- custom-resource.application-autoscaling.amazonaws.com

## Registro de recursos personalizados como destinos escalables con Auto Scaling de aplicaciones

Auto Scaling de aplicaciones requiere un destino escalable antes de que pueda crear políticas de escalado o acciones programadas para un recurso personalizado. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones. Los destinos escalables se identifican de forma única mediante la combinación de ID de recurso, dimensión escalable y espacio de nombres.

Para configurar el autoescalado mediante la AWS CLI o uno de AWS los SDK, puede usar las siguientes opciones:

- AWS CLI:

Llamada al comando [register-scalable-target](#) para un recurso personalizado. En el ejemplo siguiente se registra un recurso personalizado como destino escalable, con un recuento mínimo deseado de una unidad de capacidad y un recuento máximo deseado de 10 unidades de capacidad. El archivo `custom-resource-id.txt` contiene una cadena que identifica el ID del recurso, que representa la ruta al recurso personalizado a través de su punto de enlace de Amazon API Gateway.

```
aws application-autoscaling register-scalable-target \  
  --service-namespace custom-resource \  
  --scalable-dimension custom-resource:ResourceType:Property \  
  --resource-id file://~/custom-resource-id.txt \  
  --min-capacity 1 \  
  --max-capacity 10
```

Contenido de `custom-resource-id.txt`:

```
https://example.execute-api.us-west-2.amazonaws.com/prod/  
scalableTargetDimensions/1-23456789
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

- AWS SDK:

Llame a la operación [RegisterScalableTarget](#) y proporcione ResourceId, ScalableDimension, ServiceNamespace, MinCapacity y MaxCapacity como parámetros.

## Recursos relacionados

Si acaba de empezar a utilizar Application Auto Scaling, puede encontrar información adicional útil sobre cómo escalar sus recursos personalizados en la siguiente documentación:

[GitHubepositorio](#)

# Configure los recursos de Application Auto Scaling mediante AWS CloudFormation

Application Auto Scaling está integrado con AWS CloudFormation un servicio que le ayuda a modelar y configurar sus AWS recursos para que pueda dedicar menos tiempo a crear y administrar sus recursos e infraestructura. Cree una plantilla que describa todos los AWS recursos que desee y los AWS CloudFormation aprovisiona y configura automáticamente.

Cuando la utilice AWS CloudFormation, podrá reutilizar la plantilla para configurar los recursos de Application Auto Scaling de forma coherente y repetida. Describa sus recursos una vez y, a continuación, aprovisiona los mismos recursos una y otra vez en varias Cuentas de AWS regiones.

## Application Auto Scaling y AWS CloudFormation plantillas

Para aprovisionar y configurar los recursos de Auto Scaling de aplicaciones y sus servicios relacionados, debe entender las [plantillas de AWS CloudFormation](#). Las plantillas son archivos de texto con formato JSON o YAML. Estas plantillas describen los recursos que desea aprovisionar en sus AWS CloudFormation pilas. Si no estás familiarizado con JSON o YAML, puedes usar AWS CloudFormation Designer para ayudarte a empezar con AWS CloudFormation las plantillas. Para obtener más información, consulte [¿Qué es Designer de AWS CloudFormation ?](#) en la Guía del usuario de AWS CloudFormation .

Al crear una plantilla de pila para recursos de Auto Scaling de aplicaciones, debe proporcionar lo siguiente:

- Un espacio de nombres para el servicio de destino (por ejemplo, **appstream**). Consulta la [AWS::ApplicationAutoScaling::ScalableTarget](#) referencia para obtener los espacios de nombres de los servicios.
- Una dimensión escalable de destino asociada al recurso de destino (por ejemplo, **appstream:fleet:DesiredCapacity**). Consulte la [AWS::ApplicationAutoScaling::ScalableTarget](#) referencia para obtener dimensiones escalables.
- Un ID de recurso para el recurso de destino (por ejemplo, **fleet/sample-fleet**). Consulte la [AWS::ApplicationAutoScaling::ScalableTarget](#) referencia para obtener información sobre la sintaxis y ejemplos de identificadores de recursos específicos.
- Un rol vinculado al servicio para el recurso de destino (por ejemplo, **arn:aws:iam::012345678910:role/aws-service-role/**

**`appstream.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_AppStreamFleet`**). Consulte la tabla [Referencia de funciones vinculadas al servicio ARN](#) para obtener los ARN de rol.

Para obtener más información acerca de los recursos de Application Auto Scaling, consulte la referencia de [Application Auto Scaling](#) en la Guía del usuario de AWS CloudFormation .

## Fragmentos de ejemplo de plantilla

Puedes encontrar ejemplos de fragmentos para incluirlos en las AWS CloudFormation plantillas en las siguientes secciones de la Guía del AWS CloudFormation usuario:

- Para ver ejemplos de políticas de escalado y acciones programadas, consulte [Configurar los recursos de Application Auto Scaling con AWS CloudFormation](#).
- Para ver más ejemplos de políticas de escalado, consulte [AWS::ApplicationAutoScaling::ScalingPolicy](#).

## Más información sobre AWS CloudFormation

Para obtener más información AWS CloudFormation, consulte los siguientes recursos:

- [AWS CloudFormation](#)
- [AWS CloudFormation Guía del usuario](#)
- [AWS CloudFormation Referencia de la API](#)
- [Guía del usuario de la interfaz de la línea de comandos de AWS CloudFormation](#)

# Escalado programado para Auto Scaling de aplicaciones

Con el escalado programado, puede configurar el escalado automático para su aplicación en función de cambios de carga predecibles mediante la creación de acciones programadas que aumenten o disminuyan la capacidad en momentos específicos. Esto le permite escalar su aplicación de forma proactiva para adaptarla a los cambios de carga predecibles.

Por ejemplo, supongamos que experimenta un patrón de tráfico semanal regular en el que la carga aumenta a mitad de semana y disminuye hacia el final de la semana. Puede configurar un programa de escalado en Auto Scaling de aplicaciones que se alinee con este patrón:

- El miércoles por la mañana, una acción programada aumentará la capacidad al aumentar la capacidad mínima previamente establecida del objetivo escalable.
- El viernes por la noche, otra acción programada reduce la capacidad al disminuir la capacidad máxima previamente establecida del objetivo escalable.

Estas acciones de escalado programadas le permiten optimizar los costes y el rendimiento. Su aplicación tiene la capacidad suficiente para gestionar los picos de tráfico a mitad de semana, pero no aprovisiona en exceso la capacidad innecesaria en otros momentos.

Puede combinar el escalado programado y las políticas de escalado para obtener los beneficios de enfoques tanto proactivos como reactivos al escalado. Después de ejecutar una acción de escalado programado, la política de escalado puede seguir tomando decisiones sobre si desea ampliar la capacidad. Esto le ayuda a garantizar que tiene capacidad suficiente para controlar la carga de su aplicación. Mientras la aplicación se escala para adaptarse a la demanda, la capacidad actual debe estar dentro de la capacidad mínima y máxima establecida por la acción programada.

## Contenido

- [Cómo funciona el escalado programado para Application Auto Scaling](#)
- [Cree acciones programadas para Application Auto Scaling mediante el AWS CLI](#)
- [Describa el escalado programado para Application Auto Scaling mediante el AWS CLI](#)
- [Programe acciones de escalado recurrentes con Application Auto Scaling](#)
- [Desactivar el escalado programado para un destino escalable](#)
- [Elimine una acción programada para Application Auto Scaling mediante el AWS CLI](#)

# Cómo funciona el escalado programado para Application Auto Scaling

En este tema se describe cómo funciona el escalado programado y se presentan las consideraciones clave que debe comprender para utilizarlo de forma eficaz.

## Contenido

- [Funcionamiento](#)
- [Consideraciones](#)
- [Comandos de uso frecuente para la creación, la administración y la eliminación de acciones programadas](#)
- [Recursos relacionados](#)
- [Limitaciones](#)

## Funcionamiento

Para utilizar el escalado programado, cree acciones programadas que indican a Auto Scaling de aplicaciones que realice actividades de escalado en momentos específicos. Cuando crea una acción programada, especifica el destino escalable, cuándo se debe producir la actividad de escalado, la capacidad mínima y la capacidad máxima. Puede crear acciones programadas que realizan el escalado de forma puntual o periódica.

Cuando llega la hora especificada, Auto Scaling de aplicaciones escala según los nuevos valores de capacidad, comparando la capacidad actual con la capacidad mínima y la capacidad máxima especificada.

- Si la capacidad actual es inferior a la capacidad mínima especificada, Auto Scaling de aplicaciones escala horizontalmente hasta la capacidad mínima especificada.
- Si la capacidad actual es superior a la capacidad máxima especificada, Auto Scaling de aplicaciones reduce horizontalmente hasta la capacidad máxima especificada.

## Consideraciones

Cuando cree una acción programada, tenga en cuenta lo siguiente:

- Una acción programada configura el `MinCapacity` y `MaxCapacity` en lo especificado por la acción programada en la fecha y horas especificadas. La solicitud puede incluir opcionalmente solo uno de estos tamaños. Por ejemplo, puede crear una acción programada con solo la especificación de la capacidad mínima. Sin embargo, en algunos casos, debe incluir ambos tamaños para asegurarse de que la nueva capacidad mínima no es superior a la capacidad máxima o que la nueva capacidad máxima no sea inferior a la capacidad mínima.
- De forma predeterminada, las programaciones recurrentes se establecen en Hora universal coordinada (UTC). Puede cambiar la zona hora para que se corresponda con la zona horaria local o con una zona horaria de otra parte de la red. Cuando se especifica una zona horaria que observa el horario de verano, la acción se ajusta automáticamente al horario de verano (DST). Para obtener más información, consulte [Programa acciones de escalado recurrentes con Application Auto Scaling](#).
- Puede desactivar temporalmente el escalado programado para un destino escalable. Esto evita que las acciones programadas estén activas sin tener que eliminarlas. Podrá reanudar el escalado programado cuando desee volver a utilizarlo. Para obtener más información, consulte [Suspensión y reanudación del escalado para Application Auto Scaling](#).
- El orden en el cual las acciones programadas se ejecutan está garantizado para el mismo destino escalable, pero no para las acciones programadas en los distintos destinos escalables.
- Para completar correctamente una acción programada, el recurso especificado debe estar en un estado escalable en el servicio de destino. En caso contrario, ocurrirá un error en la solicitud y se devolverá un mensaje de error, por ejemplo, `Resource Id [ActualResourceId] is not scalable. Reason: The status of all DB instances must be 'available' or 'incompatible-parameters'`.
- Debido a la naturaleza distribuida de Auto Scaling de aplicaciones y de los servicios de destino, el retraso entre el momento en que la acción programada se activa y el momento en que el servicio de destino realiza la acción de escalado puede ser de unos segundos. Como las acciones programadas se ejecutan en el orden en el que se especifican, las acciones programadas con horas de inicio cercanas pueden tardar más en ejecutarse.

## Comandos de uso frecuente para la creación, la administración y la eliminación de acciones programadas

Los comandos comúnmente utilizados para trabajar con escalado de programación incluyen:

- [register-scalable-target](#) registrar AWS o personalizar los recursos como objetivos escalables (un recurso que Application Auto Scaling puede escalar) y suspender y reanudar el escalado.
- [put-scheduled-action](#) para añadir o modificar acciones programadas para un objetivo escalable existente.
- [describe-scaling-activities](#) para devolver información sobre la ampliación de las actividades en una AWS región.
- [describe-scheduled-actions](#) para devolver información sobre las acciones programadas en una AWS región.
- [delete-scheduled-action](#) para eliminar una acción programada.

## Recursos relacionados

Para ver un ejemplo detallado del uso del escalado programado, consulta la entrada del blog [Cómo programar la simultaneidad AWS Lambda aprovisionada para los picos de uso recurrentes](#) en el blog de AWS informática.

Para obtener información sobre cómo crear acciones programadas para grupos Auto Scaling, consulte [Escalado programado para Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

## Limitaciones

A continuación, se describen las limitaciones que se aplican cuando se utiliza escalado programado:

- Los nombres de las acciones programadas deben ser únicos por destino escalable.
- Auto Scaling de aplicaciones no proporciona precisión de segundo nivel en expresiones de programación. La mejor resolución al utilizar una expresión cron es 1 minuto.
- El destino escalable no puede ser un clúster de Amazon MSK. Amazon MSK no soporta escalado programado.
- El acceso a la consola para ver, agregar, actualizar o eliminar acciones programadas en recursos escalables depende del recurso que utilice. Para obtener más información, consulte [Servicios de AWS que puede usar con Application Auto Scaling](#).

# Cree acciones programadas para Application Auto Scaling mediante el AWS CLI

Los siguientes ejemplos muestran cómo crear acciones programadas mediante el AWS CLI [put-scheduled-action](#) comando. Cuando especifique la nueva capacidad, puede indicar la capacidad mínima, la capacidad máxima o ambas.

Estos ejemplos utilizan objetivos escalables para algunos de los servicios que se integran con Application Auto Scaling. Para usar un objetivo escalable diferente, especifique su espacio de nombres en `--service-namespace`, su dimensión escalable en `--scalable-dimension` y su ID de recurso en `--resource-id`

Cuando utilice el AWS CLI, recuerde que sus comandos se ejecutan en la Región de AWS configuración para su perfil. Si desea ejecutar los comandos en otra región, cambie la región predeterminada de su perfil o utilice el parámetro `--region` con el comando.

## Ejemplos

- [Creación de una acción programada que se produce solo una vez](#)
- [Crear de una acción programada que se ejecuta en un intervalo recurrente](#)
- [Creación de una acción programada que se ejecute en una programación recurrente](#)
- [Crear una acción programada puntual que especifica una zona horaria](#)
- [Creación de una acción programada recurrente que especifique una zona horaria](#)

## Creación de una acción programada que se produce solo una vez

Para escalar automáticamente el objetivo escalable una sola vez, en una fecha y hora especificadas, utilice la opción `--schedule "at(yyyy-mm-ddThh:mm:ss)"`.

Example Ejemplo: escalado horizontal una sola vez

A continuación se muestra un ejemplo de creación de una acción programada para escalar horizontalmente la capacidad en una fecha y hora específicas.

En la fecha y la hora se especifica para `--schedule` (22:00 horas UTC del 31 de marzo de 2021), si el valor especificado para `MinCapacity` es superior a la capacidad actual, Auto Scaling de aplicaciones escala horizontalmente a `MinCapacity`.

Linux, macOS o Unix

```
aws application-autoscaling put-scheduled-action --service-namespace custom-resource \
--scalable-dimension custom-resource:ResourceType:Property \
--resource-id file://~/custom-resource-id.txt \
--scheduled-action-name scale-out \
--schedule "at(2021-03-31T22:00:00)" \
--scalable-target-action MinCapacity=3
```

## Windows

```
aws application-autoscaling put-scheduled-action --service-namespace custom-resource ^
--scalable-dimension custom-resource:ResourceType:Property ^
--resource-id file://~/custom-resource-id.txt ^
--scheduled-action-name scale-out ^
--schedule "at(2021-03-31T22:00:00)" ^
--scalable-target-action MinCapacity=3
```

Cuando se ejecuta esta acción programada, si la capacidad máxima es menor que el valor especificado para la capacidad mínima, debe especificar una nueva capacidad mínima y máxima, y no sólo la capacidad mínima.

Example Ejemplo: reducción horizontal una sola vez

A continuación se muestra un ejemplo de creación de una acción programada para reducir horizontalmente la capacidad en una fecha y hora específicas.

En la fecha y la hora se especifica para las `--schedule` (22:30 horas UTC del 31 de marzo de 2021), si el valor especificado en `MaxCapacity` es inferior a la capacidad actual, Auto Scaling de aplicaciones reduce horizontalmente a `MaxCapacity`.

Linux, macOS o Unix

```
aws application-autoscaling put-scheduled-action --service-namespace custom-resource \
--scalable-dimension custom-resource:ResourceType:Property \
--resource-id file://~/custom-resource-id.txt \
--scheduled-action-name scale-in \
--schedule "at(2021-03-31T22:30:00)" \
--scalable-target-action MinCapacity=0,MaxCapacity=0
```

## Windows

```
aws application-autoscaling put-scheduled-action --service-namespace custom-resource ^
--scalable-dimension custom-resource:ResourceType:Property ^
```

```
--resource-id file://~/custom-resource-id.txt ^
--scheduled-action-name scale-in ^
--schedule "at(2021-03-31T22:30:00)" ^
--scalable-target-action MinCapacity=0,MaxCapacity=0
```

## Crear de una acción programada que se ejecuta en un intervalo recurrente

Para programar el escalado en un intervalo recurrente, utilice la opción `--schedule "rate(value unit)"`. El valor debe ser un número entero positivo. La unidad puede ser `minute`, `minutes`, `hour`, `hours`, `day` o `days`. Para obtener más información, consulta [Expresiones de tarifas](#) en la Guía del EventBridge usuario de Amazon.

A continuación se muestra un ejemplo de una acción programada que utiliza una expresión de tasa.

Según el horario especificado (cada 5 horas comenzando el 30 de enero de 2021 a las 24:00 horas UTC y terminando el 31 de enero de 2021 a las 22:00 horas UTC), si el valor especificado para `MinCapacity` es superior a la capacidad actual, Auto Scaling de aplicaciones escala horizontalmente a `MinCapacity`. Si el valor especificado en `MaxCapacity` es inferior a la capacidad actual, Auto Scaling de aplicaciones reduce horizontalmente a `MaxCapacity`.

Linux, macOS o Unix

```
aws application-autoscaling put-scheduled-action --service-namespace ecs \
--scalable-dimension ecs:service:DesiredCount \
--resource-id service/my-cluster/my-service \
--scheduled-action-name my-recurring-action \
--schedule "rate(5 hours)" \
--start-time 2021-01-30T12:00:00 \
--end-time 2021-01-31T22:00:00 \
--scalable-target-action MinCapacity=3,MaxCapacity=10
```

Windows

```
aws application-autoscaling put-scheduled-action --service-namespace ecs ^
--scalable-dimension ecs:service:DesiredCount ^
--resource-id service/my-cluster/my-service ^
--scheduled-action-name my-recurring-action ^
--schedule "rate(5 hours)" ^
--start-time 2021-01-30T12:00:00 ^
--end-time 2021-01-31T22:00:00 ^
--scalable-target-action MinCapacity=3,MaxCapacity=10
```

## Creación de una acción programada que se ejecute en una programación recurrente

Para programar el escalado con una programación recurrente, utilice la opción `--schedule` `"cron(fields)"`. Para obtener más información, consulte [Programe acciones de escalado recurrentes con Application Auto Scaling](#).

A continuación se muestra un ejemplo de una acción programada que utiliza una expresión cron.

Cuando llega la programación especificada (todos los días a las 9:00 horas UTC), si el valor especificado en `MinCapacity` es superior a la capacidad actual, Auto Scaling de aplicaciones escala horizontalmente a `MinCapacity`. Si el valor especificado en `MaxCapacity` es inferior a la capacidad actual, Auto Scaling de aplicaciones reduce horizontalmente a `MaxCapacity`.

Linux, macOS o Unix

```
aws application-autoscaling put-scheduled-action --service-namespace appstream \  
  --scalable-dimension appstream:fleet:DesiredCapacity \  
  --resource-id fleet/sample-fleet \  
  --scheduled-action-name my-recurring-action \  
  --schedule "cron(0 9 * * ? *)" \  
  --scalable-target-action MinCapacity=10,MaxCapacity=50
```

Windows

```
aws application-autoscaling put-scheduled-action --service-namespace appstream ^  
  --scalable-dimension appstream:fleet:DesiredCapacity ^  
  --resource-id fleet/sample-fleet ^  
  --scheduled-action-name my-recurring-action ^  
  --schedule "cron(0 9 * * ? *)" ^  
  --scalable-target-action MinCapacity=10,MaxCapacity=50
```

## Crear una acción programada puntual que especifica una zona horaria

De forma predeterminada, las acciones programadas se establecen en la zona horaria UTC. Para especificar una zona horaria diferente, incluya la opción `--timezone` y especifique el nombre canónico para la zona horaria (`America/New_York`, por ejemplo). Para obtener más información, consulta <https://www.joda.org/joda-time/timezones.html>, donde encontrarás información sobre las zonas horarias de la IANA que se admiten al llamar `put-scheduled-action`.

A continuación, se muestra un ejemplo que utiliza la opción `--timezone` al crear una acción programada para escalar la capacidad en una fecha y hora específicas.

En la fecha y la hora se especifica para `--schedule` (17:00 horas hora local del 31 de enero de 2021), si el valor especificado para `MinCapacity` es superior a la capacidad actual, Auto Scaling de aplicaciones escala horizontalmente a `MinCapacity`. Si el valor especificado en `MaxCapacity` es inferior a la capacidad actual, Auto Scaling de aplicaciones reduce horizontalmente a `MaxCapacity`.

Linux, macOS o Unix

```
aws application-autoscaling put-scheduled-action --service-namespace comprehend \  
  --scalable-dimension comprehend:document-classifier-endpoint:DesiredInferenceUnits \  
  --resource-id arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/  
EXAMPLE \  
  --scheduled-action-name my-one-time-action \  
  --schedule "at(2021-01-31T17:00:00)" --timezone "America/New_York" \  
  --scalable-target-action MinCapacity=1,MaxCapacity=3
```

Windows

```
aws application-autoscaling put-scheduled-action --service-namespace comprehend ^  
  --scalable-dimension comprehend:document-classifier-endpoint:DesiredInferenceUnits ^  
  --resource-id arn:aws:comprehend:us-west-2:123456789012:document-classifier-endpoint/  
EXAMPLE ^  
  --scheduled-action-name my-one-time-action ^  
  --schedule "at(2021-01-31T17:00:00)" --timezone "America/New_York" ^  
  --scalable-target-action MinCapacity=1,MaxCapacity=3
```

## Creación de una acción programada recurrente que especifique una zona horaria

A continuación, se muestra un ejemplo que utiliza la opción `--timezone` al crear una acción programada recurrente para escalar la capacidad. Para obtener más información, consulte [Programe acciones de escalado recurrentes con Application Auto Scaling](#).

Cuando llega la programación especificada (todos los días de lunes a viernes a las 18:00 horas hora local), si el valor especificado en `MinCapacity` es superior a la capacidad actual, Auto Scaling de aplicaciones escala horizontalmente a `MinCapacity`. Si el valor especificado en `MaxCapacity` es inferior a la capacidad actual, Auto Scaling de aplicaciones reduce horizontalmente a `MaxCapacity`.

Linux, macOS o Unix

```
aws application-autoscaling put-scheduled-action --service-namespace lambda \  
--scalable-dimension lambda:function:ProvisionedConcurrency \  
--resource-id function:my-function:BLUE \  
--scheduled-action-name my-recurring-action \  
--schedule "cron(0 18 ? * MON-FRI *)" --timezone "Etc/GMT+9" \  
--scalable-target-action MinCapacity=10,MaxCapacity=50
```

## Windows

```
aws application-autoscaling put-scheduled-action --service-namespace lambda ^  
--scalable-dimension lambda:function:ProvisionedConcurrency ^  
--resource-id function:my-function:BLUE ^  
--scheduled-action-name my-recurring-action ^  
--schedule "cron(0 18 ? * MON-FRI *)" --timezone "Etc/GMT+9" ^  
--scalable-target-action MinCapacity=10,MaxCapacity=50
```

## Describa el escalado programado para Application Auto Scaling mediante el AWS CLI

Estos AWS CLI comandos de ejemplo describen las actividades de escalado y las acciones programadas utilizando recursos de los servicios que se integran con Application Auto Scaling. Para un objetivo escalable diferente, especifique su espacio de nombres en `--service-namespace`, su dimensión escalable en `--scalable-dimension` y su ID de recurso en `--resource-id`.

Cuando utilice el AWS CLI, recuerde que sus comandos se ejecutan en la Región de AWS configuración de su perfil. Si desea ejecutar los comandos en otra región, cambie la región predeterminada de su perfil o utilice el parámetro `--region` con el comando.

### Ejemplos

- [Describa las actividades de escalado de un servicio](#)
- [Describa las acciones programadas para un servicio](#)
- [Describa las acciones programadas para un objetivo escalable](#)

## Describa las actividades de escalado de un servicio

Para ver las actividades de escalado de todos los destinos escalables de un espacio de nombres de servicio específico, utilice el [describe-scaling-activities](#) comando.

En el siguiente ejemplo se recuperan las actividades de escalado asociadas con el espacio de nombres de servicio dynamodb.

Linux, macOS o Unix

```
aws application-autoscaling describe-scaling-activities --service-namespace dynamodb
```

Windows

```
aws application-autoscaling describe-scaling-activities --service-namespace dynamodb
```

Salida

Si el comando se ejecuta correctamente, devuelve un resultado similar al siguiente.

```
{
  "ScalingActivities": [
    {
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
      "Description": "Setting write capacity units to 10.",
      "ResourceId": "table/my-table",
      "ActivityId": "4d1308c0-bbcf-4514-a673-b0220ae38547",
      "StartTime": 1561574415.086,
      "ServiceNamespace": "dynamodb",
      "EndTime": 1561574449.51,
      "Cause": "maximum capacity was set to 10",
      "StatusMessage": "Successfully set write capacity units to 10. Change
successfully fulfilled by dynamodb.",
      "StatusCode": "Successful"
    },
    {
      "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
      "Description": "Setting min capacity to 5 and max capacity to 10",
      "ResourceId": "table/my-table",
      "ActivityId": "f2b7847b-721d-4e01-8ef0-0c8d3bacc1c7",
      "StartTime": 1561574414.644,
      "ServiceNamespace": "dynamodb",
      "Cause": "scheduled action name my-second-scheduled-action was triggered",
      "StatusMessage": "Successfully set min capacity to 5 and max capacity to
10",
      "StatusCode": "Successful"
    },
    {
```

```

    "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
    "Description": "Setting write capacity units to 15.",
    "ResourceId": "table/my-table",
    "ActivityId": "d8ea4de6-9eaa-499f-b466-2cc5e681ba8b",
    "StartTime": 1561574108.904,
    "ServiceNamespace": "dynamodb",
    "EndTime": 1561574140.255,
    "Cause": "minimum capacity was set to 15",
    "StatusMessage": "Successfully set write capacity units to 15. Change
successfully fulfilled by dynamodb.",
    "StatusCode": "Successful"
  },
  {
    "ScalableDimension": "dynamodb:table:WriteCapacityUnits",
    "Description": "Setting min capacity to 15 and max capacity to 20",
    "ResourceId": "table/my-table",
    "ActivityId": "3250fd06-6940-4e8e-bb1f-d494db7554d2",
    "StartTime": 1561574108.512,
    "ServiceNamespace": "dynamodb",
    "Cause": "scheduled action name my-first-scheduled-action was triggered",
    "StatusMessage": "Successfully set min capacity to 15 and max capacity to
20",
    "StatusCode": "Successful"
  }
]
}

```

Para cambiar este comando de modo que recupere las actividades de escalado solo para uno de sus destinos escalables, agregue la opción `--resource-id`.

## Describa las acciones programadas para un servicio

Para describir las acciones programadas para todos los destinos escalables de un espacio de nombres de servicio específico, utilice el [describe-scheduled-actions](#) comando.

En el ejemplo siguiente se recuperan las acciones programadas asociadas con el `ec2` espacio de nombres de servicio.

Linux, macOS o Unix

```
aws application-autoscaling describe-scheduled-actions --service-namespace ec2
```

Windows

```
aws application-autoscaling describe-scheduled-actions --service-namespace ec2
```

## Salida

Si el comando se ejecuta correctamente, devuelve un resultado similar al siguiente.

```
{
  "ScheduledActions": [
    {
      "ScheduledActionName": "my-one-time-action",
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledAction:493a6261-fbb9-432d-855d-3c302c14bdb9:resource/ec2/
spot-fleet-request/sfr-107dc873-0802-4402-a901-37294EXAMPLE:scheduledActionName/my-one-
time-action",
      "ServiceNamespace": "ec2",
      "Schedule": "at(2021-01-31T17:00:00)",
      "Timezone": "America/New_York",
      "ResourceId": "spot-fleet-request/sfr-107dc873-0802-4402-
a901-37294EXAMPLE",
      "ScalableDimension": "ec2:spot-fleet-request:TargetCapacity",
      "ScalableTargetAction": {
        "MaxCapacity": 1
      },
      "CreationTime": 1607454792.331
    },
    {
      "ScheduledActionName": "my-recurring-action",
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledAction:493a6261-fbb9-432d-855d-3c302c14bdb9:resource/ec2/
spot-fleet-request/sfr-107dc873-0802-4402-a901-37294EXAMPLE:scheduledActionName/my-
recurring-action",
      "ServiceNamespace": "ec2",
      "Schedule": "rate(5 minutes)",
      "ResourceId": "spot-fleet-request/sfr-107dc873-0802-4402-
a901-37294EXAMPLE",
      "ScalableDimension": "ec2:spot-fleet-request:TargetCapacity",
      "StartTime": 1604059200.0,
      "EndTime": 1612130400.0,
      "ScalableTargetAction": {
        "MinCapacity": 3,
        "MaxCapacity": 10
      },
      "CreationTime": 1607454949.719
    }
  ]
}
```

```

    },
    {
      "ScheduledActionName": "my-one-time-action",
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledAction:4bce34c7-bb81-4ecf-b776-5c726efb1567:resource/ec2/
spot-fleet-request/sfr-40edeb7b-9ae7-44be-bef2-5c4c8EXAMPLE:scheduledActionName/my-one-
time-action",
      "ServiceNamespace": "ec2",
      "Schedule": "at(2020-12-08T9:36:00)",
      "Timezone": "America/New_York",
      "ResourceId": "spot-fleet-request/sfr-40edeb7b-9ae7-44be-
bef2-5c4c8EXAMPLE",
      "ScalableDimension": "ec2:spot-fleet-request:TargetCapacity",
      "ScalableTargetAction": {
        "MinCapacity": 1,
        "MaxCapacity": 3
      },
      "CreationTime": 1607456031.391
    }
  ]
}

```

## Describa las acciones programadas para un objetivo escalable

Para recuperar información sobre las acciones programadas para un objetivo escalable específico, añada la `--resource-id` opción al describir las acciones programadas mediante el [describe-scheduled-actions](#) comando.

Si incluye la opción `--scheduled-action-names` y especifica el nombre de una acción programada como su valor, el comando devuelve sólo la acción programada cuyo nombre es una coincidencia, como se muestra en el siguiente ejemplo.

Linux, macOS o Unix

```

aws application-autoscaling describe-scheduled-actions --service-namespace ec2 \
--resource-id spot-fleet-request/sfr-40edeb7b-9ae7-44be-bef2-5c4c8EXAMPLE \
--scheduled-action-names my-one-time-action

```

Windows

```

aws application-autoscaling describe-scheduled-actions --service-namespace ec2 ^

```

```
--resource-id spot-fleet-request/sfr-40edeb7b-9ae7-44be-bef2-5c4c8EXAMPLE ^
--scheduled-action-names my-one-time-action
```

## Salida

Si el comando se ejecuta correctamente, devuelve un resultado similar al siguiente. Si ha proporcionado más de un valor para `--scheduled-action-names`, el resultado incluye todas las acciones programadas cuyos nombres coincidan.

```
{
  "ScheduledActions": [
    {
      "ScheduledActionName": "my-one-time-action",
      "ScheduledActionARN": "arn:aws:autoscaling:us-
west-2:123456789012:scheduledAction:4bce34c7-bb81-4ecf-b776-5c726efb1567:resource/ec2/
spot-fleet-request/sfr-40edeb7b-9ae7-44be-bef2-5c4c8EXAMPLE:scheduledActionName/my-one-
time-action",
      "ServiceNamespace": "ec2",
      "Schedule": "at(2020-12-08T9:36:00)",
      "Timezone": "America/New_York",
      "ResourceId": "spot-fleet-request/sfr-40edeb7b-9ae7-44be-
bef2-5c4c8EXAMPLE",
      "ScalableDimension": "ec2:spot-fleet-request:TargetCapacity",
      "ScalableTargetAction": {
        "MinCapacity": 1,
        "MaxCapacity": 3
      },
      "CreationTime": 1607456031.391
    }
  ]
}
```

## Programe acciones de escalado recurrentes con Application Auto Scaling

### Important

Para obtener ayuda con las expresiones cron de Amazon EC2 Auto Scaling, consulte el tema [Programaciones recurrentes](#) de la Guía del usuario de Amazon EC2 Auto Scaling. Amazon

EC2 Auto Scaling utiliza la sintaxis cron tradicional en lugar de la sintaxis cron personalizada que utiliza Application Auto Scaling.

También puede crear acciones programadas que se ejecuten en una programación recurrente utilizando una expresión cron.

Para crear una programación recurrente, especifique una expresión cron y una zona horaria para describir cuándo se va a repetir esa acción programada. Los valores de zona horaria admitidos son los nombres canónicos de las zonas horarias de IANA admitidas por [Joda-Time](#) (tales como Etc/GMT+9 o Pacific/Tahiti). Opcionalmente, puede especificar una fecha y una hora para la hora de inicio, la hora de finalización o para ambas. Para ver un ejemplo de comando que utiliza el AWS CLI para crear una acción programada, consulte [Creación de una acción programada recurrente que especifique una zona horaria](#).

El formato de expresión cron admitido consta de seis campos separados por espacios en blanco: [Minutos] [Horas] [Día\_del\_mes] [Mes] [Día\_de\_la\_semana] [Año]. Por ejemplo, la expresión cron 30 6 ? \* MON \* configura una acción programada que se repite cada lunes a las 06:30 a. m. El asterisco se utiliza como comodín para coincidir con todos los valores de un campo.

Para obtener más información sobre la sintaxis cron para las acciones programadas de Application Auto Scaling, consulte la [referencia de expresiones cron](#) en la Guía EventBridge del usuario de Amazon.

Cuando cree una programación periódica, elija cuidadosamente sus horas de inicio y finalización. Tenga en cuenta lo siguiente:

- Si especifica una hora de inicio, Auto Scaling de aplicaciones realiza la acción en ese momento y, a continuación, realiza la acción basada en la programación periódica.
- Si especifica una hora de finalización, la acción deja de repetirse después de esta hora. Auto Scaling de aplicaciones no realiza un seguimiento de los valores anteriores y vuelve a los valores anteriores después de la hora de finalización.
- La hora de inicio y la hora de finalización deben estar configuradas en UTC cuando utilices el SDK AWS CLI o los AWS SDK para crear o actualizar una acción programada.

## Ejemplos

Puede consultar la siguiente tabla cuando cree una programación periódica para un destino escalable de Auto Scaling de aplicaciones. Los siguientes ejemplos muestran la sintaxis correcta para utilizar Auto Scaling de aplicaciones con el objetivo de crear o actualizar una acción programada.

Minutos	Horas	Día del mes	Mes	Día de la semana	Año	Significado
0	10	*	*	?	*	Ejecutar a las 10:00 h (UTC) todos los días
15	12	*	*	?	*	Ejecutar a las 12:15 h (UTC) todos los días
0	18	?	*	MON-FRI	*	Ejecutar a las 18:00 h (UTC) de lunes a viernes
0	8	1	*	?	*	Ejecutar a las 8:00 horas (UTC) cada primer día del mes
0/15	*	*	*	?	*	Ejecutar cada 15 minutos

Minutos	Horas	Día del mes	Mes	Día de la semana	Año	Significado
0/10	*	?	*	MON-FRI	*	Ejecutar cada 10 minutos de lunes a viernes
0/5	8-17	?	*	MON-FRI	*	Ejecutar cada 5 minutos de lunes a viernes entre las 8.00 y las 17.55 h (UTC)

### Excepción

También puede crear una expresión cron con un valor de cadena que contenga siete campos. En este caso, puede utilizar los tres primeros campos para especificar la hora de ejecución de una acción programada, incluidos los segundos. La expresión cron completa tiene los siguientes campos separados por espacios: [Segundos] [Minutos] [Horas] [Día\_del\_mes] [Mes] [Día\_de\_la\_semana] [Año]. Sin embargo, este enfoque no garantiza que la acción programada se ejecute en el segundo preciso que especifique. Además, es posible que algunas consolas de servicio no admitan el campo de segundos de una expresión cron.

## Desactivar el escalado programado para un destino escalable

Puede desactivar temporalmente el escalado programado sin eliminar las acciones programadas. Para obtener más información, consulte [Suspensión y reanudación del escalado para Application Auto Scaling](#).

Para suspender el escalado programado

Suspenda el escalado programado en un objetivo escalable mediante el [register-scalable-target](#) comando de la `--suspended-state` opción y especificando `true` el valor del `ScheduledScalingSuspended` atributo como se muestra en el siguiente ejemplo.

Linux, macOS o Unix

```
aws application-autoscaling register-scalable-target --service-namespace rds \  
  --scalable-dimension rds:cluster:ReadReplicaCount --resource-id cluster:my-db-cluster \  
  --suspended-state '{"ScheduledScalingSuspended": true}'
```

Windows

```
aws application-autoscaling register-scalable-target --service-namespace rds ^ \  
  --scalable-dimension rds:cluster:ReadReplicaCount --resource-id cluster:my-db-cluster \  
  --suspended-state "{\"ScheduledScalingSuspended\": true}"
```

Salida

Si el comando se ejecuta correctamente, devuelve el ARN del objetivo escalable. A continuación, se muestra un ejemplo del resultado.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

Para reanudar el escalado programado

Para reanudar el escalado programado, ejecute de nuevo el `register-scalable-target` comando, especificando `false` como valor para `ScheduledScalingSuspended`.

## Elimine una acción programada para Application Auto Scaling mediante el AWS CLI

Cuando ya no necesite una acción programada, puede eliminarla.

Para eliminar la acción programada

Utilice el comando [delete-scheduled-action](#). Si se ejecuta correctamente, este comando no devuelve ningún resultado.

Linux, macOS o Unix

```
aws application-autoscaling delete-scheduled-action \  
  --service-namespace ec2 \  
  --scalable-dimension ec2:spot-fleet-request:TargetCapacity \  
  --resource-id spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-37294EXAMPLE \  
  --scheduled-action-name my-recurring-action
```

Windows

```
aws application-autoscaling delete-scheduled-action ^  
  --service-namespace ec2 ^  
  --scalable-dimension ec2:spot-fleet-request:TargetCapacity ^  
  --resource-id spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-37294EXAMPLE ^  
  --scheduled-action-name my-recurring-action
```

Anular el registro del objetivo escalable

Si también ha terminado con el objetivo escalable, puede anular su registro. Use el siguiente comando [deregister-scalable-target](#). Si hay políticas de escalado o acciones programadas que aún no se hayan eliminado, este comando las eliminará. Si se ejecuta correctamente, este comando no devuelve ningún resultado.

Linux, macOS o Unix

```
aws application-autoscaling deregister-scalable-target \  
  --service-namespace ec2 \  
  --scalable-dimension ec2:spot-fleet-request:TargetCapacity \  
  --resource-id spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-37294EXAMPLE
```

Windows

```
aws application-autoscaling deregister-scalable-target ^  
  --service-namespace ec2 ^  
  --scalable-dimension ec2:spot-fleet-request:TargetCapacity ^  
  --resource-id spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-37294EXAMPLE
```

# Políticas de escalado de seguimiento de destino para Auto Scaling de aplicaciones

Una política de escalado de seguimiento de destino escalado automáticamente de aplicación en función de un valor de métrica de destino. Esto permite que su aplicación mantenga un rendimiento y una rentabilidad óptimos sin intervención manual.

Con el seguimiento de objetivo, seleccione una métrica y un valor objetivo para representar el nivel ideal de utilización promedio o rendimiento para su aplicación. Application Auto Scaling crea y administra las CloudWatch alarmas que activan los eventos de escalado cuando la métrica se desvía del objetivo. Esto es similar a cómo un termostato mantiene una temperatura objetivo.

Por ejemplo, supongamos que tiene una aplicación que actualmente se pone en la flota de spot y desea que la utilización de CPU de la flota permanezca en torno al 50 % cuando cambie la carga en la aplicación. De este modo dispone de capacidad adicional para gestionar picos de tráfico sin mantener una cantidad excesiva de recursos inactivos.

Puede satisfacer esta necesidad mediante la creación de una política de escalado de seguimiento de destino que tenga como destino una utilización media de CPU del 50 por ciento. Luego, Auto Scaling de aplicaciones se escalará horizontalmente(aumentará la capacidad) cuando la CPU supere el 50 por ciento para gestionar el aumento de carga. Se reducirá horizontalmente (reducirá la capacidad) cuando la CPU caiga por debajo del 50 por ciento para optimizar los costos durante los periodos de baja utilización.

Las políticas de seguimiento de Target eliminan la necesidad de definir manualmente CloudWatch las alarmas y los ajustes de escalado. Auto Scaling de aplicaciones gestiona esto automáticamente en función del objetivo que establezca.

Puede basar políticas de escalado de destino en métricas predefinidas o personalizadas:

- Métricas predefinidas: métricas proporcionadas por Auto Scaling de aplicaciones, como el uso promedio de la CPU o el recuento promedio de solicitudes por objetivo.
- Métricas personalizadas: puedes usar las matemáticas métricas para combinar métricas, aprovechar las métricas existentes o usar tus propias métricas personalizadas publicadas para CloudWatch ti.

Elija una métrica que cambie de forma inversamente proporcional al cambio en la capacidad de su objetivo escalable. Por lo tanto, si duplicas la capacidad, la métrica disminuye un 50 por ciento. Esto permite que los datos de las métricas activen con precisión los eventos de escalado proporcional.

## Contenido

- [Cómo funciona el escalado de seguimiento de objetivos para Application Auto Scaling](#)
- [Cree una política de escalado de seguimiento de objetivos para Application Auto Scaling mediante el AWS CLI](#)
- [Elimine una política de escalado de seguimiento de objetivos para Application Auto Scaling mediante el AWS CLI](#)
- [Creación de una política de escalado de seguimiento de destino para Application Auto Scaling con la calculadora de métricas](#)

# Cómo funciona el escalado de seguimiento de objetivos para Application Auto Scaling

En este tema se describe cómo funciona el escalado de seguimiento de objetivos y se presentan los elementos clave de una política de escalado de seguimiento de objetivos.

## Contenido

- [Funcionamiento](#)
- [Elección de métricas](#)
- [Definición del valor de destino](#)
- [Defina los periodos de recuperación](#)
- [Consideraciones](#)
- [Políticas de escalado múltiples](#)
- [Comandos de uso frecuente para la creación, administración y eliminación de políticas de escalado](#)
- [Recursos relacionados](#)
- [Limitaciones](#)

## Funcionamiento

Para utilizar el escalado de seguimiento de objetivos, debe crear una política de escalado de seguimiento de objetivos y especificar lo siguiente:

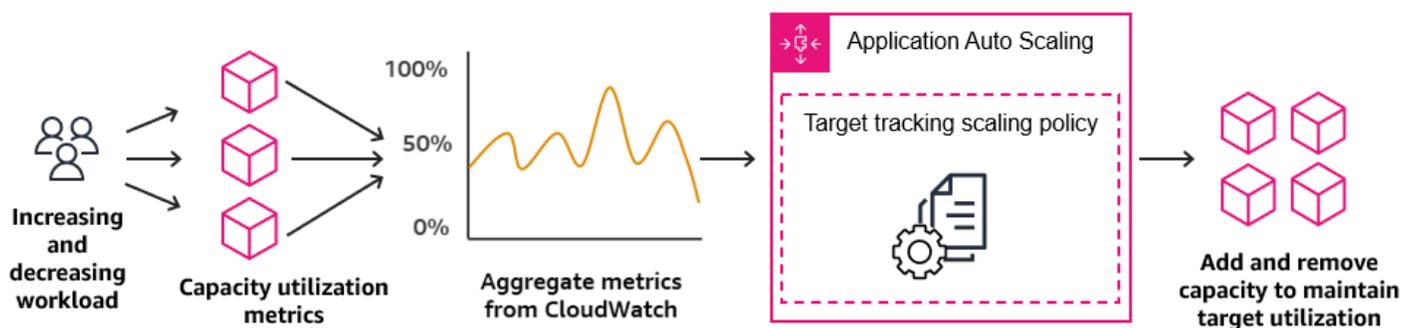
- **Métrica:** una CloudWatch métrica para realizar un seguimiento, como el uso medio de la CPU o el recuento medio de solicitudes por objetivo.
- **Valor objetivo:** el valor objetivo de la métrica, como el 50 por ciento de uso de la CPU o 1000 solicitudes por objetivo por minuto.

Application Auto Scaling crea y administra las CloudWatch alarmas que invocan la política de escalado y calcula el ajuste de escalado en función de la métrica y el valor objetivo. Amplía y reduce la capacidad en función de las necesidades para mantener la métrica en el valor objetivo especificado o en un valor próximo.

Cuando la métrica está por encima del valor objetivo, Auto Scaling de aplicaciones se amplía añadiendo capacidad para reducir la diferencia entre el valor de la métrica y el valor objetivo. Cuando la métrica está por debajo del valor objetivo, Auto Scaling de aplicaciones se amplía al eliminar la capacidad.

Las actividades de escalado se realizan con periodos de recuperación entre ellos para evitar fluctuaciones rápidas de la capacidad. Si lo desea, puede configurar los periodos de recuperación para su política de escalado.

En el siguiente diagrama se muestra información general de cómo funciona una política de escalado de seguimiento de destino cuando se completa la configuración.



Tenga en cuenta que la política de escalado de seguimiento de destino es más agresiva a la hora de agregar capacidad cuando la utilización aumenta que a la hora de eliminar la capacidad cuando la utilización disminuye. Por ejemplo, si la métrica especificada de la política alcanza su valor

objetivo, la política supone que la aplicación ya está muy cargada. Por lo tanto, responde agregando capacidad proporcional al valor de métrica lo más rápido posible. Cuanto mayor sea la métrica, mayor será la capacidad agregada.

Cuando la métrica cae por debajo del valor objetivo, la política espera que vuelva a aumentar la utilización. En este caso, ralentiza el escalado al eliminar la capacidad solo cuando el uso supera un umbral que está lo suficientemente por debajo del valor de destino (generalmente más del 10 % menos) para que se considere que se ha ralentizado el uso. La intención de este comportamiento más conservador es garantizar que la eliminación de capacidad solo se produzca cuando la aplicación ya no tenga demanda al mismo nivel alto que antes.

## Elección de métricas

Puede crear políticas de escalado de seguimiento de destino con métricas predefinidas o personalizadas.

Al crear una política de escalado de seguimiento de destino con un tipo de métrica predefinido, debe elegir una métrica de la lista de métricas predefinidas en [Métricas predefinidas para políticas de escalado de seguimiento de destino](#).

Tenga en cuenta las siguientes consideraciones al elegir una métrica:

- No todas las métricas personalizadas funcionan para el seguimiento de destino. La métrica debe ser una métrica de utilización válida y describir el nivel de actividad de un destino escalable. El valor de la métrica debe aumentar o disminuir proporcionalmente a la capacidad del destino escalable, de forma que los datos de la métrica se puedan utilizar para escalar proporcionalmente el destino escalable.
- Para utilizar la métrica `ALBRequestCountPerTarget`, debe especificar el parámetro `ResourceLabel` para identificar el grupo de destino asociado con la métrica.
- Cuando una métrica emite valores 0 reales a CloudWatch (por ejemplo, `ALBRequestCountPerTarget`), Application Auto Scaling puede escalar a 0 cuando no hay tráfico en la aplicación durante un período prolongado de tiempo. Para que su objetivo escalable se reduzca horizontalmente a 0 cuando no se enrutan solicitudes, la capacidad mínima del objetivo escalable debe establecerse a 0.
- En lugar de publicar métricas nuevas para utilizarlas en su política de escalado, puede utilizar las matemáticas métricas para combinar las métricas existentes. Para obtener más información, consulte [Creación de una política de escalado de seguimiento de destino para Application Auto Scaling con la calculadora de métricas](#).

- Para ver si un servicio que esté utilizando admite la especificación de métricas personalizadas en la consola del servicio, consulte la documentación correspondiente a dicho servicio.
- Le recomendamos que utilice métricas que estén disponibles en intervalos de un minuto para ayudarlo a escalar más rápido en respuesta a los cambios de uso. El seguimiento de objetivos evaluará las métricas agregadas con un grado de detalle de un minuto para todas las métricas predefinidas y personalizadas, pero es posible que la métrica subyacente publique datos con menos frecuencia. Por ejemplo, todas las métricas de Amazon EC2 se envían en intervalos de cinco minutos de forma predeterminada, pero se pueden configurar en un minuto (lo que se conoce como monitoreo detallado). Esta elección depende de los servicios individuales. La mayoría trata de utilizar el intervalo más corto posible.

## Definición del valor de destino

Al crear una política de escalado de seguimiento de destino, debe especificar un valor de destino. El valor de destino representa el uso o el rendimiento promedio óptimo para su aplicación. Para usar los recursos de manera rentable, establezca el valor objetivo lo más alto posible con un búfer razonable para aumentos inesperados de tráfico. Cuando la aplicación se escala horizontalmente de manera óptima para un flujo de tráfico normal, el valor de la métrica real debe ser igual al valor de destino, o estar justo por debajo de él.

Cuando una política de escalado se basa en el rendimiento, como el recuento de solicitudes por objetivo para un equilibrador de carga de aplicación, E/S de red u otras métricas de recuento, el valor objetivo representa el rendimiento promedio óptimo de una sola entidad (por ejemplo, un único objetivo del grupo objetivo del equilibrador de carga de aplicación), para un periodo de un minuto.

## Defina los periodos de recuperación

Si lo desea, puede definir periodos de recuperación en su política de escalado de seguimiento de destinos.

Un periodo de recuperación especifica la cantidad de tiempo que la política de escalado espera a que una actividad de escalado anterior surta efecto.

Existen dos tipos de periodos de recuperación:

- Con el periodo de recuperación de escalado ascendente, la intención es efectuar un escalo ascendente de forma continua (pero no excesivamente). Después de que Application Auto Scaling efectúe de forma correcta el escalado horizontal con una política de escalado, comienza a calcular

el tiempo de recuperación. Una política de escalado no volverá a aumentar la capacidad deseada a menos que se desencadene un escalado horizontal mayor o finalice el periodo de recuperación. Mientras el periodo de recuperación del escalado ascendente esté en vigor, la capacidad agregada por la actividad inicial de escalado ascendente se considerará parte de la capacidad deseada para la siguiente actividad de escalado ascendente.

- Con el periodo de recuperación de reducción horizontal, la intención es reducir horizontalmente con precaución para proteger la disponibilidad de la aplicación, de modo que las actividades de reducir horizontalmente se bloqueen hasta que haya transcurrido el periodo de recuperación de reducción horizontal. No obstante, si otra alarma desencadena una actividad de escalado ascendente durante el periodo de recuperación de escalado descendente, Application Auto Scaling realiza inmediatamente un escalado ascendente del destino. En este caso, el periodo de recuperación de la reducción horizontal se detiene y no se completa.

Cada periodo de recuperación se mide en segundos y solo se aplica a las actividades de escalado relacionadas con políticas de escalado. Durante un periodo de recuperación, cuando una acción programada comienza a la hora programada, puede desencadenar una actividad de escalado inmediatamente sin esperar a que finalice el periodo de recuperación.

Puede comenzar con los valores predeterminados, que se pueden ajustar más adelante. Por ejemplo, es posible que deba aumentar un periodo de recuperación para evitar que la política de escalado de seguimiento de destino sea demasiado agresiva con respecto a los cambios que se producen en cortos periodos de tiempo.

### Valores predeterminados

Application Auto Scaling proporciona un valor predeterminado de 600 para los grupos de ElastiCache replicación y un valor predeterminado de 300 para los siguientes destinos escalables:

- AppStream Flotas 2.0
- Clústeres de base de datos de Aurora
- Servicios de ECS
- Clústeres de Neptune
- SageMaker variantes de punto final
- SageMaker componentes de inferencia
- SageMaker simultaneidad aprovisionada sin servidor
- Spot Fleets

- Pool de WorkSpaces
- Recursos personalizados

Para todos los demás destinos escalables, el valor predeterminado es 0 o nulo:

- Puntos de conexión de reconocedor de identidades y clasificación de documentos de Amazon Comprehend
- Tablas de DynamoDB e índices secundarios globales
- Tablas de Amazon Keyspaces
- Simultaneidad aprovisionada de Lambda
- Almacenamiento de bróker de Amazon MSK

Los valores nulos se tratan igual que los valores cero cuando Application Auto Scaling evalúa el periodo de recuperación.

Puede actualizar cualquiera de los valores predeterminados, incluidos los valores nulos, para establecer sus propios periodos de recuperación.

## Consideraciones

Las siguientes consideraciones se aplican al trabajar con las políticas de escalado de seguimiento de destino:

- No cree, edite ni elimine las CloudWatch alarmas que se utilizan con una política de escalado de seguimiento de objetivos. Application Auto Scaling crea y administra las CloudWatch alarmas asociadas a sus políticas de escalado y seguimiento de objetivos y las elimina cuando ya no son necesarias.
- Si a la métrica le faltan puntos de datos, el estado de la CloudWatch alarma cambia a `INSUFFICIENT_DATA`. Cuando esto ocurre, Application Auto Scaling no puede escalar su objetivo escalable hasta que se encuentren nuevos puntos de datos. Para obtener más información, consulta [Cómo configurar el modo en que CloudWatch las alarmas tratan los datos faltantes](#) en la Guía del CloudWatch usuario de Amazon.
- Si la métrica se presenta de forma dispersa por diseño, las matemáticas métricas pueden resultar útiles. Por ejemplo, para usar los valores más recientes, utilice la función `FILL(m1, REPEAT)`, donde `m1` es la métrica.

- Es posible que haya diferencias entre el valor objetivo y los puntos de datos de la métrica real. Esto se debe a que Auto Scaling de aplicaciones siempre actúa de forma conservadora y redondea hacia arriba o hacia abajo a la hora de determinar la cantidad de capacidad que debe agregar o quitar. Con esto se evita que se agregue capacidad insuficiente o se elimine demasiada capacidad. Sin embargo, en el caso de un destino escalable con poca capacidad, podría parecer que los puntos de datos de la métrica están muy alejados del valor objetivo.

En el caso de un destino escalable con mayor capacidad, cuando se agrega o elimina capacidad, la diferencia entre el valor de destino y los datos reales de la métrica es menor.

- En las políticas de escalado de seguimiento de destino, se presupone que el escalado ascendente se realiza cuando la métrica está por encima del valor objetivo. No puede utilizar una política de escalado de seguimiento de destino si la métrica especificada está por debajo del valor objetivo.

## Políticas de escalado múltiples

Puede tener varias políticas de escalado de seguimiento de destino para un destino escalable, siempre que cada una de ellas utilice una métrica diferente. El objetivo de Auto Scaling de aplicaciones siempre es dar prioridad a la disponibilidad, por lo que su comportamiento varía en rol de si las políticas de seguimiento de destino están listas para el escalado horizontal o reducción horizontal. Realizará un escalado ascendente del destino escalable si cualquiera de las políticas de seguimiento de destino está lista para el escalado ascendente, pero solo realizará el escalado descendente si todas las políticas de seguimiento de destino (que tienen la parte de escalado descendente habilitada) están listas para el escalado descendente.

Si varias políticas de escalado indican al destino escalable que escale horizontalmente o reduzca horizontalmente al mismo tiempo, Auto Scaling de aplicaciones se escala en función de la política que proporciona la mayor capacidad para escalar horizontalmente y reducir horizontalmente. Esto brinda una mayor flexibilidad para abordar diferentes situaciones y garantizar que siempre haya capacidad suficiente para procesar las cargas de trabajo.

Puede deshabilitar la parte de reducción horizontal de una política de escalado de seguimiento de destino para utilizar un método para reducir horizontalmente diferente al que utiliza para escalar horizontalmente. Por ejemplo, puede utilizar una política de escalado por pasos para el escalado descendente mientras utiliza una política de escalado de seguimiento de destino para el escalado ascendente.

Sin embargo, recomendamos precaución al utilizar políticas de escalado de seguimiento de destino con políticas de escalado por pasos, ya que los conflictos entre estas políticas pueden provocar un

comportamiento no deseado. Por ejemplo, si la política de escalado por pasos inicia una actividad de reducción horizontal antes de que la política de seguimiento de destino esté lista para la reducción horizontal, la actividad de reducción horizontal no se bloqueará. Una vez completada la actividad de escalado descendente, la política de seguimiento de destino podría indicar al destino escalable que vuelva a realizar el escalado ascendente.

Para las cargas de trabajo de naturaleza cíclica, también tiene la opción de automatizar los cambios de capacidad según una programación mediante escalado programado. Para cada acción programada, se pueden definir un nuevo valor mínimo de capacidad y otro nuevo máximo. Estos valores constituyen los límites de la política de escalado. La combinación de escalado programado y escalado de seguimiento de destino puede contribuir a reducir el impacto de un fuerte aumento de los niveles de utilización, cuando se necesita capacidad inmediatamente.

## Comandos de uso frecuente para la creación, administración y eliminación de políticas de escalado

Los comandos comúnmente utilizados para trabajar con políticas de escalado incluyen:

- [register-scalable-target](#) registrar AWS o personalizar los recursos como objetivos escalables (un recurso que Application Auto Scaling puede escalar) y suspender y reanudar el escalado.
- [put-scaling-policy](#) para añadir o modificar políticas de escalado para un objetivo escalable existente.
- [describe-scaling-activities](#) para devolver información sobre el escalamiento de las actividades en una AWS región.
- [describe-scaling-policies](#) para devolver información sobre las políticas de escalado en una AWS región.
- [delete-scaling-policy](#) para eliminar una política de escalado.

## Recursos relacionados

Para obtener más información sobre cómo crear políticas de escalado de seguimiento de destino para grupos de Auto Scaling, consulte [Políticas de escalado de seguimiento de destino para Amazon EC2 Auto Scaling](#) en la Guía de usuario de Amazon EC2 Auto Scaling.

## Limitaciones

Las siguientes son limitaciones al usar la política de escalado de seguimiento de destino:

- El destino escalable no puede ser un clúster de Amazon EMR. Las políticas de escalado de seguimiento de destino no son compatibles con Amazon EMR.
- Cuando un clúster de Amazon MSK es el destino escalable, reducir horizontalmente se desactiva y no se puede habilitar.
- No puede utilizar las operaciones de la RegisterScalableTarget PutScalingPolicy API para actualizar un plan de AWS Auto Scaling escalado.
- El acceso a la consola para ver, agregar, actualizar o eliminar políticas de escalado de seguimiento de destino en recursos escalables depende del recurso que utilice. Para obtener más información, consulte [Servicios de AWS que puede usar con Application Auto Scaling](#).

## Cree una política de escalado de seguimiento de objetivos para Application Auto Scaling mediante el AWS CLI

En este ejemplo, se utilizan AWS CLI comandos para crear una política de almacenamiento objetivo para una flota de Amazon EC2 Spot. Para un objetivo escalable diferente, especifique su espacio de nombres en `--service-namespace`, su dimensión escalable en `y` su ID de `--scalable-dimension` recurso en `--resource-id`

Cuando utilice AWS CLI, recuerde que sus comandos se ejecutan en la Región de AWS configuración de su perfil. Si desea ejecutar los comandos en otra región, cambie la región predeterminada de su perfil o utilice el parámetro `--region` con el comando.

### Tareas

- [Paso 1: Registrar un objetivo escalable](#)
- [Paso 2: Crear una política de escalado de seguimiento de destino](#)
- [Paso 3: Describa las políticas de escalado y seguimiento de objetivos](#)

## Paso 1: Registrar un objetivo escalable

Si aún no lo ha hecho, registre el destino escalable. Utilice el [register-scalable-target](#) comando para registrar un recurso específico en el servicio de destino como destino escalable. En el ejemplo siguiente se registra una solicitud de flota de spot con Auto Scaling de aplicaciones. Auto Scaling de aplicaciones puede escalar el número de instancias de flota de spot en un mínimo de 2 instancias y un máximo de 10. Reemplace cada *marcador de posición de entrada del usuario* con información propia.

## Linux, macOS o Unix

```
aws application-autoscaling register-scalable-target --service-namespace ec2 \  
--scalable-dimension ec2:spot-fleet-request:TargetCapacity \  
--resource-id spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \  
--min-capacity 2 --max-capacity 10
```

## Windows

```
aws application-autoscaling register-scalable-target --service-namespace ec2 ^  
--scalable-dimension ec2:spot-fleet-request:TargetCapacity ^  
--resource-id spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE ^  
--min-capacity 2 --max-capacity 10
```

## Salida

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable. A continuación, se muestra un ejemplo del resultado.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

## Paso 2: Crear una política de escalado de seguimiento de destino

Para crear una política de escalado y seguimiento de objetivos, puede utilizar los siguientes ejemplos como ayuda para empezar.

Para crear una política de escalado de seguimiento de destino

1. Usa el siguiente `cat` comando para almacenar un valor objetivo para tu política de escalado y una especificación métrica predefinida en un archivo JSON nombrado `config.json` en tu directorio principal. El siguiente es un ejemplo de configuración de seguimiento de objetivos que mantiene la utilización media de la CPU en un 50 por ciento.

```
$ cat ~/config.json  
{  
  "TargetValue": 50.0,  
  "PredefinedMetricSpecification":  
    {
```

```

    "PredefinedMetricType": "EC2SpotFleetRequestAverageCPUUtilization"
  }
}

```

Para obtener más información, consulte la Referencia [PredefinedMetricSpecification](#) de la API Application Auto Scaling.

Como alternativa, puede usar una métrica personalizada para escalar creando una especificación de métrica personalizada y agregando valores para cada parámetro desde CloudWatch. El siguiente es un ejemplo de configuración de seguimiento de objetivos que mantiene la utilización media de la métrica especificada en 100.

```

$ cat ~/config.json
{
  "TargetValue": 100.0,
  "CustomizedMetricSpecification":{
    "MetricName": "MyUtilizationMetric",
    "Namespace": "MyNamespace",
    "Dimensions": [
      {
        "Name": "MyOptionalMetricDimensionName",
        "Value": "MyOptionalMetricDimensionValue"
      }
    ],
    "Statistic": "Average",
    "Unit": "Percent"
  }
}

```

Para obtener más información, consulte la Referencia [CustomizedMetricSpecification](#) de la API Application Auto Scaling.

- Utilice el siguiente [put-scaling-policy](#) comando, junto con el config.json archivo que creó, para crear una política de escalado denominada `cpu50-target-tracking-scaling-policy`.

Linux, macOS o Unix

```

aws application-autoscaling put-scaling-policy --service-namespace ec2 \
  --scalable-dimension ec2:spot-fleet-request:TargetCapacity \
  --resource-id spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
  --policy-name cpu50-target-tracking-scaling-policy --policy-type
TargetTrackingScaling \

```

```
--target-tracking-scaling-policy-configuration file://config.json
```

## Windows

```
aws application-autoscaling put-scaling-policy --service-namespace ec2 ^  
--scalable-dimension ec2:spot-fleet-request:TargetCapacity ^  
--resource-id spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE ^  
--policy-name cpu50-target-tracking-scaling-policy --policy-type  
TargetTrackingScaling ^  
--target-tracking-scaling-policy-configuration file://config.json
```

## Salida

Si se ejecuta correctamente, este comando devuelve los ARN y los nombres de las dos CloudWatch alarmas creadas en su nombre. A continuación, se muestra un ejemplo del resultado.

```
{  
  "PolicyARN": "arn:aws:autoscaling:region:account-  
id:scalingPolicy:policy-id:resource/ec2/spot-fleet-request/sfr-73fbd2ce-  
aa30-494c-8788-1cee4EXAMPLE:policyName/cpu50-target-tracking-scaling-policy",  
  "Alarms": [  
    {  
      "AlarmARN": "arn:aws:cloudwatch:region:account-id:alarm:TargetTracking-  
spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE-AlarmHigh-d4f0770c-  
b46e-434a-a60f-3b36d653feca",  
      "AlarmName": "TargetTracking-spot-fleet-request/sfr-73fbd2ce-  
aa30-494c-8788-1cee4EXAMPLE-AlarmHigh-d4f0770c-b46e-434a-a60f-3b36d653feca"  
    },  
    {  
      "AlarmARN": "arn:aws:cloudwatch:region:account-id:alarm:TargetTracking-  
spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE-AlarmLow-1b437334-  
d19b-4a63-a812-6c67aaf2910d",  
      "AlarmName": "TargetTracking-spot-fleet-request/sfr-73fbd2ce-  
aa30-494c-8788-1cee4EXAMPLE-AlarmLow-1b437334-d19b-4a63-a812-6c67aaf2910d"  
    }  
  ]  
}
```

## Paso 3: Describa las políticas de escalado y seguimiento de objetivos

Puede describir todas las políticas de escalado del espacio de nombres del servicio especificado mediante el siguiente [describe-scaling-policies](#) comando.

```
aws application-autoscaling describe-scaling-policies --service-namespace ec2
```

Puede filtrar los resultados a solo las políticas de escalado de seguimiento de destino mediante el parámetro `--query`. Para obtener más información acerca de syntax para `query`, consulte [Control de la salida de comandos de la AWS CLI](#) en la Guía del usuario de AWS Command Line Interface .

Linux, macOS o Unix

```
aws application-autoscaling describe-scaling-policies --service-namespace ec2 \  
--query 'ScalingPolicies[?PolicyType==`TargetTrackingScaling`]'
```

Windows

```
aws application-autoscaling describe-scaling-policies --service-namespace ec2 ^  
--query "ScalingPolicies[?PolicyType==`TargetTrackingScaling`]"
```

Salida

A continuación, se muestra un ejemplo del resultado.

```
[  
  {  
    "PolicyARN": "PolicyARN",  
    "TargetTrackingScalingPolicyConfiguration": {  
      "PredefinedMetricSpecification": {  
        "PredefinedMetricType": "EC2SpotFleetRequestAverageCPUUtilization"  
      },  
      "TargetValue": 50.0  
    },  
    "PolicyName": "cpu50-target-tracking-scaling-policy",  
    "ScalableDimension": "ec2:spot-fleet-request:TargetCapacity",  
    "ServiceNamespace": "ec2",  
    "PolicyType": "TargetTrackingScaling",  
    "ResourceId": "spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",  
    "Alarms": [  
      {
```

```

        "AlarmARN": "arn:aws:cloudwatch:region:account-id:alarm:TargetTracking-
spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE-AlarmHigh-d4f0770c-
b46e-434a-a60f-3b36d653feca",
        "AlarmName": "TargetTracking-spot-fleet-request/sfr-73fbd2ce-
aa30-494c-8788-1cee4EXAMPLE-AlarmHigh-d4f0770c-b46e-434a-a60f-3b36d653feca"
    },
    {
        "AlarmARN": "arn:aws:cloudwatch:region:account-id:alarm:TargetTracking-
spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE-AlarmLow-1b437334-
d19b-4a63-a812-6c67aaf2910d",
        "AlarmName": "TargetTracking-spot-fleet-request/sfr-73fbd2ce-
aa30-494c-8788-1cee4EXAMPLE-AlarmLow-1b437334-d19b-4a63-a812-6c67aaf2910d"
    }
],
"CreationTime": 1515021724.807
}
]

```

## Elimine una política de escalado de seguimiento de objetivos para Application Auto Scaling mediante el AWS CLI

Cuando haya terminado con una política de escalado de seguimiento de objetivos, puede eliminarla mediante el [delete-scaling-policy](#) comando.

El siguiente comando elimina la política de escalado de seguimiento de destino especificada para la solicitud de flota de spot indicada. También elimina las CloudWatch alarmas que Application Auto Scaling creó en su nombre.

Linux, macOS o Unix

```

aws application-autoscaling delete-scaling-policy --service-namespace ec2 \
--scalable-dimension ec2:spot-fleet-request:TargetCapacity \
--resource-id spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
--policy-name cpu50-target-tracking-scaling-policy

```

Windows

```

aws application-autoscaling delete-scaling-policy --service-namespace ec2 ^
--scalable-dimension ec2:spot-fleet-request:TargetCapacity ^
--resource-id spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE ^
--policy-name cpu50-target-tracking-scaling-policy

```

# Creación de una política de escalado de seguimiento de destino para Application Auto Scaling con la calculadora de métricas

Con la matemática métrica, puede consultar múltiples CloudWatch métricas y usar expresiones matemáticas para crear nuevas series temporales basadas en estas métricas. Puede visualizar las series temporales resultantes en la CloudWatch consola y añadirlas a los paneles. Para obtener más información sobre las matemáticas métricas, consulte [Uso de las matemáticas métricas](#) en la Guía del CloudWatch usuario de Amazon.

Las siguientes consideraciones se aplican a las expresiones de la calculadora de métricas:

- Puede consultar cualquier CloudWatch métrica disponible. Cada métrica es una combinación única de nombre de métrica, espacio de nombres y cero o más dimensiones.
- Puede usar cualquier operador aritmético (+ - \*/^), función estadística (como AVG o SUM) u otra función compatible. CloudWatch
- Puede utilizar tanto las métricas como los resultados de otras expresiones matemáticas en las fórmulas de la expresión matemática.
- Todas las expresiones utilizadas en la especificación de una métrica deben devolver en última instancia una única serie temporal.
- Puede comprobar que una expresión matemática métrica es válida mediante la CloudWatch consola o la API. CloudWatch [GetMetricData](#)

## Temas

- [Ejemplo: cola de tareas pendientes de Amazon SQS por tarea](#)
- [Limitaciones](#)

## Ejemplo: cola de tareas pendientes de Amazon SQS por tarea

Para calcular la cola de tareas pendientes de Amazon SQS por tarea, se toma el número aproximado de mensajes disponibles para recuperar de la cola y se divide por el número de tareas de Amazon ECS ejecutándose en el servicio. Para obtener más información, consulte [Auto Scaling de Amazon Elastic Container Service \(ECS\) mediante métricas personalizadas](#) en el blog de AWS informática.

La lógica de la expresión es la siguiente:

`sum of (number of messages in the queue)/(number of tasks that are currently in the RUNNING state)`

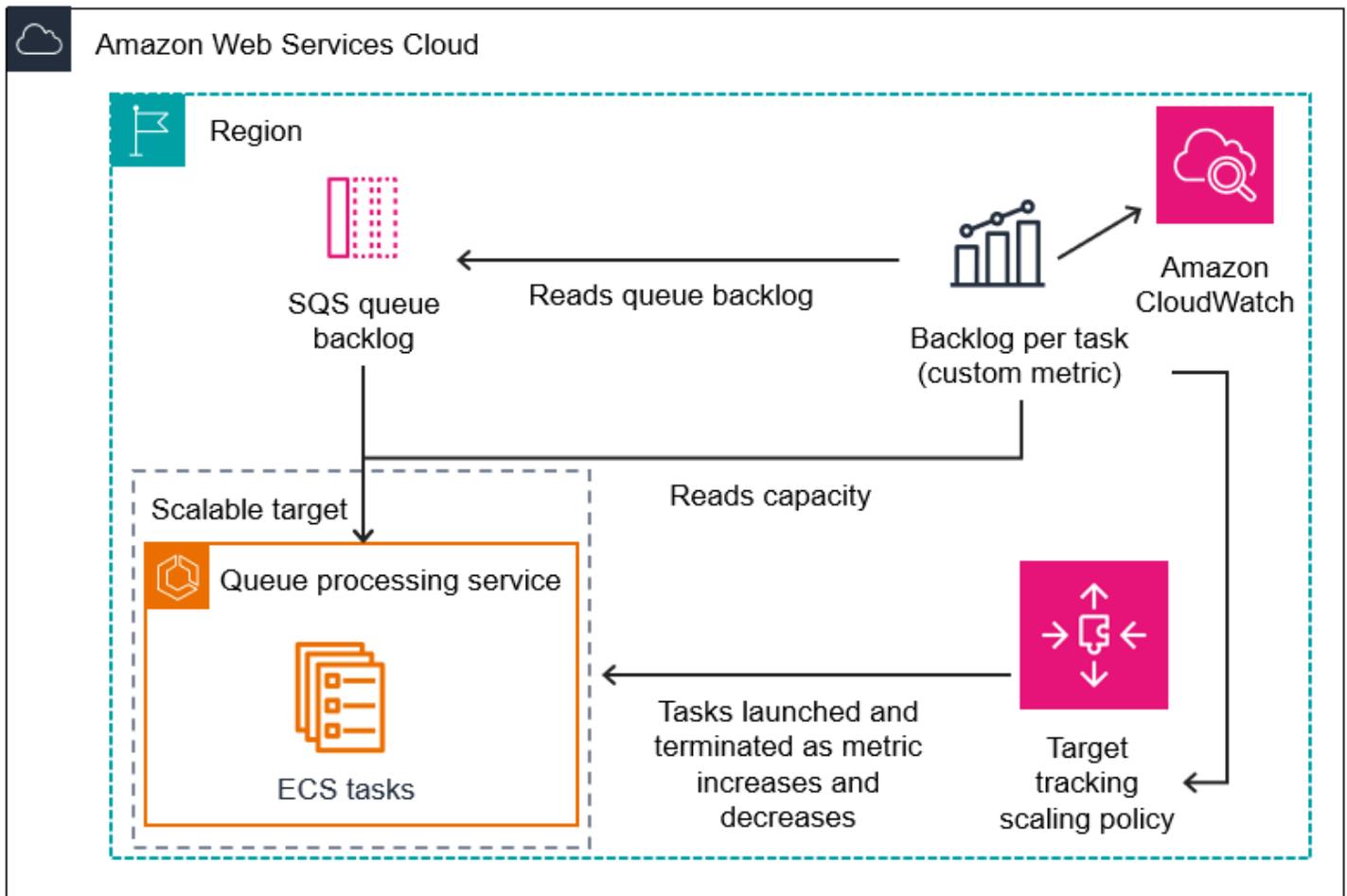
A continuación, la información de sus CloudWatch métricas es la siguiente.

ID	CloudWatch métrica	Estadística	Período
m1	ApproximateNumberOfMessagesVisible	Sum	1 minuto
m2	RunningTaskCount	Media	1 minuto

Su ID de cálculo de métrica y expresión son los siguientes.

ID	Expression
e1	<code>(m1)/(m2)</code>

El siguiente diagrama ilustra la arquitectura de esta métrica:



Para utilizar esta calculadora de métricas para crear una política de escalado de seguimiento de destino (AWS CLI)

1. Guarde la expresión de la calculadora de métricas como parte de una especificación métrica personalizada en un archivo JSON denominado `config.json`.

Utilice el siguiente ejemplo como ayuda para comenzar. Reemplace cada *marcador de posición de entrada del usuario* con información propia.

```
{
  "CustomizedMetricSpecification": {
    "Metrics": [
      {
        "Label": "Get the queue size (the number of messages waiting to be processed)",
        "Id": "m1",
        "MetricStat": {
```

```

        "Metric": {
            "MetricName": "ApproximateNumberOfMessagesVisible",
            "Namespace": "AWS/SQS",
            "Dimensions": [
                {
                    "Name": "QueueName",
                    "Value": "my-queue"
                }
            ]
        },
        "Stat": "Sum"
    },
    "ReturnData": false
},
{
    "Label": "Get the ECS running task count (the number of currently
running tasks)",
    "Id": "m2",
    "MetricStat": {
        "Metric": {
            "MetricName": "RunningTaskCount",
            "Namespace": "ECS/ContainerInsights",
            "Dimensions": [
                {
                    "Name": "ClusterName",
                    "Value": "my-cluster"
                },
                {
                    "Name": "ServiceName",
                    "Value": "my-service"
                }
            ]
        },
        "Stat": "Average"
    },
    "ReturnData": false
},
{
    "Label": "Calculate the backlog per instance",
    "Id": "e1",
    "Expression": "m1 / m2",
    "ReturnData": true
}
]

```

```

    },
    "TargetValue": 100
  }

```

Para obtener más información, consulte la Referencia [TargetTrackingScalingPolicyConfiguration](#) de la API Application Auto Scaling.

### Note

Los siguientes son algunos recursos adicionales que pueden ayudarle a encontrar nombres de métricas, espacios de nombres, dimensiones y estadísticas para CloudWatch las métricas:

- Para obtener información sobre las métricas disponibles para AWS los servicios, consulta [AWS los servicios que publican CloudWatch métricas](#) en la Guía del CloudWatch usuario de Amazon.
- [Para obtener el nombre, el espacio de nombres y las dimensiones exactos \(si corresponde\) de una CloudWatch métrica con el AWS CLI, consulta list-metrics.](#)

2. Para crear esta política, ejecute el [put-scaling-policy](#) comando con el archivo JSON como entrada, como se muestra en el siguiente ejemplo.

```

aws application-autoscaling put-scaling-policy --policy-name sqs-backlog-target-tracking-scaling-policy \
  --service-namespace ecs --scalable-dimension ecs:service:DesiredCount --resource-id service/my-cluster/my-service \
  --policy-type TargetTrackingScaling --target-tracking-scaling-policy-configuration file://config.json

```

Si se ejecuta correctamente, este comando devuelve el nombre de recurso de Amazon (ARN) de la política y los ARN de las dos CloudWatch alarmas creadas en su nombre.

```

{
  "PolicyARN": "arn:aws:autoscaling:us-west-2:012345678910:scalingPolicy:8784a896-b2ba-47a1-b08c-27301cc499a1:resource/ecs/service/my-cluster/my-service:policyName/sqs-backlog-target-tracking-scaling-policy",
  "Alarms": [
    {

```

```
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:TargetTracking-service/my-cluster/my-service-
AlarmHigh-9bc77b56-0571-4276-ba0f-d4178882e0a0",
        "AlarmName": "TargetTracking-service/my-cluster/my-service-
AlarmHigh-9bc77b56-0571-4276-ba0f-d4178882e0a0"
    },
    {
        "AlarmARN": "arn:aws:cloudwatch:us-
west-2:012345678910:alarm:TargetTracking-service/my-cluster/my-service-
AlarmLow-9b6ad934-6d37-438e-9e05-02836ddcbdc4",
        "AlarmName": "TargetTracking-service/my-cluster/my-service-
AlarmLow-9b6ad934-6d37-438e-9e05-02836ddcbdc4"
    }
]
}
```

### Note

Si este comando arroja un error, asegúrese de haber actualizado la versión AWS CLI local a la última versión.

## Limitaciones

- El tamaño máximo de solicitud es 50 KB. Este es el tamaño total de la carga útil de la solicitud de [PutScalingPolicy](#) API cuando se utilizan las matemáticas métricas en la definición de la política. Si supera este límite, Application Auto Scaling rechaza la solicitud.
- Los siguientes servicios no se admiten cuando se utilizan métricas matemáticas con las políticas de escalado de seguimiento de destino:
  - Amazon Keyspaces (para Apache Cassandra)
  - DynamoDB
  - Amazon EMR
  - Amazon MSK
  - Amazon Neptune

# Políticas de escalado por pasos para Auto Scaling de aplicaciones

Una política de escalado escalonado escala la capacidad de la aplicación en incrementos predefinidos en función de CloudWatch las alarmas. Puede definir políticas de escalado independientes para gestionar el escalado horizontal (aumento de la capacidad) y la reducción horizontal (reducción de la capacidad) cuando se supere el umbral de una alarma.

Con las políticas de escalado escalonado, puede crear y gestionar las CloudWatch alarmas que invocan el proceso de escalado. Cuando se infringe una alarma, Auto Scaling de aplicaciones inicia la política de escalado asociada a esa alarma.

La política de escalado por pasos escala la capacidad mediante un conjunto de ajustes, conocidos como ajustes escalonados. La magnitud del ajuste varía en función del tamaño de la interrupción de la alarma.

- Si la infracción supera el primer umbral, Auto Scaling de aplicaciones aplicará el ajuste del primer paso.
- Si la infracción supera el segundo umbral, Auto Scaling de aplicaciones aplicará el ajuste del segundo paso, y así sucesivamente.

Esto permite que la política de escalado responda adecuadamente a los cambios menores y mayores en la métrica de alarma.

La política continuará respondiendo a las interrupciones adicionales de la alarma, incluso mientras se esté realizando una actividad de escalado. Esto significa que Auto Scaling de aplicaciones evaluará todas las interrupciones de alarma a medida que se produzcan. Se utiliza un periodo de recuperación para protegerse contra el sobreescalamiento debido a que se producen varias interrupciones de alarma en rápida sucesión.

Al igual que el seguimiento de objetivos, el escalado por pasos puede ayudar a escalar automáticamente la capacidad de la aplicación a medida que se producen cambios en el tráfico. Sin embargo, las políticas de seguimiento de objetivos suelen ser más fáciles de implementar y gestionar para satisfacer necesidades de escalamiento constantes.

## Objetivos escalables compatibles

Puede utilizar políticas de escalado por pasos con los siguientes objetivos escalables:

- AppStream Flotas 2.0
- Clústeres de base de datos de Aurora
- Servicios de ECS
- Clústeres de EMR
- SageMaker variantes de punto final
- SageMaker componentes de inferencia
- SageMaker simultaneidad aprovisionada sin servidor
- Spot Fleets
- Recursos personalizados

## Contenido

- [Cómo funciona el escalado por pasos para Application Auto Scaling](#)
- [Cree una política de escalado escalonado para Application Auto Scaling mediante el AWS CLI](#)
- [Describa las políticas de escalado por pasos para Application Auto Scaling mediante el AWS CLI](#)
- [Elimine una política de escalado por pasos para Application Auto Scaling mediante el AWS CLI](#)

# Cómo funciona el escalado por pasos para Application Auto Scaling

En este tema se describe cómo funciona el escalado escalonado y se presentan los elementos clave de una política de escalado escalonado.

## Contenido

- [Funcionamiento](#)
- [Ajustes de pasos](#)
- [Tipos de ajuste de escalado](#)
- [Periodo de recuperación](#)
- [Comandos de uso frecuente para la creación, administración y eliminación de políticas de escalado](#)
- [Consideraciones](#)
- [Recursos relacionados](#)

- [Acceso a la consola](#)

## Funcionamiento

Para usar el escalado escalonado, debe crear una CloudWatch alarma que supervise una métrica para su objetivo escalable. Defina la métrica, el valor límite y el número de periodos de evaluación que determinan una interrupción de la alarma. También debe crear una política de escalado por pasos que defina cómo escalar la capacidad cuando se supere el umbral de alarma y asociarla a su objetivo escalable.

Añada los ajustes escalonados a la política. Puede definir diferentes ajustes escalonados en función del tamaño de la infracción de la alarma. Por ejemplo:

- Escale horizontalmente la capacidad en 10 unidades si la métrica de alarma alcanza el 60 por ciento
- Escale horizontalmente la capacidad en 30 unidades si la métrica de alarma alcanza el 75 por ciento
- Escale horizontalmente la capacidad en 40 unidades si la métrica de alarma alcanza el 85 por ciento

Cuando se supere el umbral de alarma durante el número especificado de periodos de evaluación, Auto Scaling de aplicaciones aplicará los ajustes escalonados definidos en la política. Los ajustes pueden continuar en caso de que se produzcan nuevas infracciones de alarma hasta que se restablezca el estado de alarma. OK

Las actividades de escalado se realizan con periodos de recuperación entre ellos para evitar fluctuaciones rápidas de la capacidad. Si lo desea, puede configurar los periodos de recuperación para su política de escalado.

## Ajustes de pasos

Cuando se crea una política de escalado por pasos, se especifican uno o varios ajustes de pasos que escalan automáticamente la capacidad del destino dinámicamente en función del tamaño de la interrupción de alarma. Cada ajuste por pasos especifica los elementos siguientes:

- El límite inferior del valor de la métrica
- El límite superior del valor de la métrica

- La cantidad que se va a escalar, en función del tipo de ajuste de escalado

CloudWatch agrega los puntos de datos de las métricas en función de la estadística de la métrica asociada a la alarma. CloudWatch Cuando se interrumpe la alarma, se invoca la política de escalado adecuada. Application Auto Scaling aplica el tipo de agregación especificado a los puntos de datos métricos más recientes de CloudWatch (a diferencia de los datos métricos sin procesar). También compara este valor agregado de la métrica con el límite inferior y superior definido por los ajustes por pasos para determinar qué ajuste por pasos debe realizar.

Usted especifica los límites superiores e inferiores en relación con el umbral de interrupción. Por ejemplo, supongamos que ha creado una CloudWatch alarma y una política de escalado horizontal para cuando la métrica supere el 50 por ciento. A continuación, ha creado una segunda alarma y una política de reducción horizontal para cuando la métrica esté por debajo del 50 por ciento. Realizó una serie de ajustes escalonados con un tipo de ajuste `PercentChangeInCapacity` para cada política:

Ejemplo: ajustes de pasos para la política de escalado ascendente

Límite inferior	Límite superior	Ajuste
0	10	0
10	20	10
20	nulo	30

Ejemplo: ajustes por pasos de la política de reducción horizontal

Límite inferior	Límite superior	Ajuste
-10	0	0
-20	-10	-10
nulo	-20	-30

Esto crea la siguiente configuración de escalado.

Metric value



Ahora supongamos que usa esta configuración de escalado en los que la capacidad es 10. En los puntos siguientes, se resume el comportamiento de la configuración de escalado en relación con la capacidad del destino escalable:

- La capacidad original se mantiene mientras que el valor agregado de la métrica sea superior a 40 e inferior a 60.
- Si el valor de la métrica llega a 60, Auto Scaling de aplicaciones aumenta la capacidad del destino escalable en 1 hasta alcanzar un total de 11. Este valor se basa en el segundo ajuste por pasos de la política de escalado ascendente (añadir el 10 por ciento de 10). Una vez agregada la nueva capacidad, Auto Scaling de aplicaciones aumenta la capacidad actual a 11. Si el valor de la métrica aumenta hasta 70 incluso después de este aumento de la capacidad, Auto Scaling de aplicaciones incrementa la capacidad de destino en 3, lo que hace un total de 14. Este valor se basa en el tercer ajuste por pasos de la política de escalado ascendente (añadir un 30 por ciento de 11, lo que da 3,3, que, al redondear, se queda en 3).
- Si el valor de la métrica llega a 40, Auto Scaling de aplicaciones reduce la capacidad del destino escalable en 1 hasta 13, en rol del segundo ajuste de paso de la política de reducir horizontalmente (quitar el 10 por ciento de 14, 1,4, redondeado a 1). Si el valor de la métrica disminuye hasta 30 incluso después de esta reducción de capacidad, Auto Scaling de aplicaciones reduce la capacidad objetivo en 3 hasta 10, en rol del tercer ajuste de paso de la política de reducir horizontalmente (agregar el 30 por ciento de 13, 3,9, redondeado a 3).

Cuando especifique los ajustes por pasos de la política de escalado, tenga en cuenta lo siguiente:

- Los intervalos de los ajustes por pasos no se pueden solapar ni contener huecos.
- Solo puede haber un ajuste por pasos con un límite inferior nulo (infinito negativo). Si un ajuste por pasos tiene un límite inferior negativo, debe haber un ajuste por pasos con un límite inferior nulo.
- Solo puede haber un ajuste por pasos con un límite superior nulo (infinito positivo). Si un ajuste por pasos tiene un límite superior positivo, debe haber un ajuste por pasos con un límite superior nulo.
- Los límites superior e inferior no pueden ser nulos en el mismo ajuste por pasos.

- Si el valor de la métrica es superior al umbral de infracción, el límite inferior es inclusivo y el límite superior es exclusivo. Si el valor de la métrica es inferior al umbral de infracción, el límite inferior es exclusivo y el límite superior es inclusivo.

## Tipos de ajuste de escalado

Puede definir una política de escalado que realice la acción de escalado idónea en función del tipo de ajuste de escalado elegido. Puede especificar el tipo de ajuste como un porcentaje de la capacidad actual del objetivo escalable o en números absolutos.

Auto Scaling de aplicaciones admite los siguientes tipos de ajuste para las políticas de escalado por pasos:

- **ChangeInCapacity**—Aumente o disminuya la capacidad actual del objetivo escalable en el valor especificado. Un valor positivo aumenta la capacidad y un valor negativo reduce la capacidad. Por ejemplo, si la capacidad actual es 3 y el ajuste es 5, Auto Scaling de aplicaciones agrega 5 a la capacidad, lo que suma un total de 8.
- **ExactCapacity**—Cambie la capacidad actual del objetivo escalable al valor especificado. Especifique un valor positivo con este tipo de ajuste. Por ejemplo, si la capacidad actual es 3 y el ajuste es 5, Auto Scaling de aplicaciones cambia la capacidad a 5.
- **PercentChangeInCapacity**—Aumente o disminuya la capacidad actual del objetivo escalable en el porcentaje especificado. Un valor positivo aumenta la capacidad y un valor negativo reduce la capacidad. Por ejemplo, si la capacidad actual es 10 y el ajuste es del 10 %, Auto Scaling de aplicaciones agrega 1 a la capacidad, lo que da un total de 11.

Si el valor resultante no es un entero, Auto Scaling de aplicaciones lo redondea como se indica a continuación:

- Los valores mayores que 1 se redondean al valor inferior. Por ejemplo, 12.7 se redondea a 12.
- Los valores comprendidos entre 0 y 1 se redondean a 1. Por ejemplo, .67 se redondea a 1.
- Los valores comprendidos entre 0 y -1 se redondean a -1. Por ejemplo, -.58 se redondea a -1.
- Los valores menores que -1 se redondean al valor superior. Por ejemplo, -6.67 se redondea a -6.

Con **PercentChangeInCapacity**, también puede especificar la cantidad mínima a escalar mediante el **MinAdjustmentMagnitude** parámetro. Suponga, por ejemplo, que crea una política que añada un 25 por ciento y especifica una cantidad mínima de 2. Si el destino escalable tiene una

capacidad de 4 y se ejecuta la política de escalado, el 25 por ciento de 4 es 1. Sin embargo, puesto que ha especificado un incremento mínimo de 2, Auto Scaling de aplicaciones agrega 2.

## Periodo de recuperación

Si lo desea, puede definir un periodo de recuperación en su política de escalado por pasos.

Un periodo de recuperación especifica la cantidad de tiempo que la política de escalado espera a que una actividad de escalado anterior surta efecto.

Hay dos formas de planificar el uso de los periodos de recuperación para una configuración de escalado por pasos:

- Con el periodo de recuperación de escalado horizontal, la intención es escalar horizontalmente de forma continua (pero no excesivamente). Después de que Application Auto Scaling efectúe de forma correcta el escalado horizontal con una política de escalado, comienza a calcular el tiempo de recuperación. Una política de escalado no volverá a aumentar la capacidad deseada a menos que se desencadene un escalado horizontal mayor o finalice el periodo de recuperación. Mientras el periodo de recuperación del escalado ascendente esté en vigor, la capacidad agregada por la actividad inicial de escalado ascendente se considerará parte de la capacidad deseada para la siguiente actividad de escalado ascendente.
- Con el periodo de recuperación de las políticas de reducción horizontal, la intención es reducir horizontalmente con precaución para proteger la disponibilidad de la aplicación, de modo que las actividades de reducir horizontalmente se bloqueen hasta que haya transcurrido el periodo de recuperación de la reducción horizontal. No obstante, si otra alarma desencadena una actividad de escalado ascendente durante el periodo de recuperación de escalado descendente, Application Auto Scaling realiza inmediatamente un escalado ascendente del destino. En este caso, el periodo de recuperación de la reducción horizontal se detiene y no se completa.

Por ejemplo, cuando se produce un pico de tráfico, se activa una alarma y Application Auto Scaling agrega capacidad automáticamente para ayudar a gestionar el aumento de carga. Si se establece un periodo de recuperación para la política de escalado horizontal, cuando la alarma activa la política para aumentar la capacidad en 2, la actividad de escalado se realiza correctamente y se inicia el periodo de recuperación del escalado horizontal. Si una alarma vuelve a activarse durante el periodo de recuperación, pero con un ajuste por pasos más estricto que 3, el aumento de 2 se considerará parte de la capacidad actual. Por lo tanto, solo se añade 1 a la capacidad. Esto permite escalar más

rápido que esperar a que caduque el tiempo de recuperación, pero sin agregar más capacidad de la que se necesita.

El período de recuperación se mide en segundos y solo se aplica a las actividades de escalado relacionadas con políticas de escalado. Durante un periodo de recuperación, cuando una acción programada comienza a la hora programada, puede desencadenar una actividad de escalado inmediatamente sin esperar a que finalice el periodo de recuperación.

El valor predeterminado es 300 si no se especifica ningún valor.

## Comandos de uso frecuente para la creación, administración y eliminación de políticas de escalado

Los comandos comúnmente utilizados para trabajar con políticas de escalado incluyen:

- [register-scalable-target](#) registrar AWS o personalizar los recursos como objetivos escalables (un recurso que Application Auto Scaling puede escalar) y suspender y reanudar el escalado.
- [put-scaling-policy](#) para añadir o modificar políticas de escalado para un objetivo escalable existente.
- [describe-scaling-activities](#) para devolver información sobre el escalamiento de las actividades en una AWS región.
- [describe-scaling-policies](#) para devolver información sobre las políticas de escalado en una AWS región.
- [delete-scaling-policy](#) para eliminar una política de escalado.

## Consideraciones

Las siguientes consideraciones se aplican al trabajar con las políticas de escalado por pasos:

- Considere si puede predecir los ajustes escalonados de la aplicación con la precisión suficiente como para utilizar la escala por pasos. Si la métrica de escalado aumenta o reduce en proporción a la capacidad del destino escalable, le recomendamos que, en su lugar, utilice una política de escalado de seguimiento de destino. Todavía tiene la opción de usar escalado por pasos como una política adicional para una configuración más avanzada. Por ejemplo, puede configurar una respuesta más agresiva cuando se alcance un determinado nivel de utilización.
- Asegúrese de elegir un margen adecuado entre los umbrales de escalado horizontal y escalado automático para evitar oscilaciones. La fluctuación es un bucle infinito de reducción horizontal y

escalado horizontal. Es decir, si se realiza una acción de escalado, el valor de la métrica cambiaría e iniciaría otra acción de escalado en la dirección inversa.

## Recursos relacionados

Para obtener más información sobre cómo crear políticas de escalado por pasos para grupos de Auto Scaling, consulte [Pasos y políticas de escalado sencillas para Amazon EC2 Auto Scaling](#) en la Guía del usuario de Amazon EC2 Auto Scaling.

## Acceso a la consola

El acceso a la consola para ver, agregar, actualizar o eliminar políticas de escalado por pasos en recursos escalables depende del recurso que utilice. Para obtener más información, consulte [Servicios de AWS que puede usar con Application Auto Scaling](#).

## Cree una política de escalado escalonado para Application Auto Scaling mediante el AWS CLI

En este ejemplo, se utilizan AWS CLI comandos para crear una política de escalado por pasos para un servicio de Amazon ECS. Para un objetivo escalable diferente, especifique su espacio de nombres en `--service-namespace`, su dimensión escalable en `--scalable-dimension` y su ID de recurso en `--resource-id`

Cuando utilice el AWS CLI, recuerde que sus comandos se ejecutan en la Región de AWS configuración de su perfil. Si desea ejecutar los comandos en otra región, cambie la región predeterminada de su perfil o utilice el parámetro `--region` con el comando.

### Tareas

- [Paso 1: Registrar un objetivo escalable](#)
- [Paso 2: Cree una política de escalado escalonado](#)
- [Paso 3: Cree una alarma que invoque una política de escalado](#)

## Paso 1: Registrar un objetivo escalable

Si aún no lo ha hecho, registre el destino escalable. Utilice el [register-scalable-target](#) comando para registrar un recurso específico en el servicio de destino como destino escalable. En el siguiente ejemplo se registra un Servicio ECS de Amazon con Auto Scaling de aplicaciones. Auto Scaling

de aplicaciones puede escalar el número de tareas a un mínimo de 2 tareas y un máximo de 10. Reemplace cada *marcador de posición de entrada del usuario* con información propia.

Linux, macOS o Unix

```
aws application-autoscaling register-scalable-target --service-namespace ecs \  
  --scalable-dimension ecs:service:DesiredCount \  
  --resource-id service/my-cluster/my-service \  
  --min-capacity 2 --max-capacity 10
```

Windows

```
aws application-autoscaling register-scalable-target --service-namespace ecs ^  
  --scalable-dimension ecs:service:DesiredCount ^  
  --resource-id service/my-cluster/my-service ^  
  --min-capacity 2 --max-capacity 10
```

Salida

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable. A continuación, se muestra un ejemplo del resultado.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

## Paso 2: Cree una política de escalado escalonado

Para crear una política de escalado escalonado para su objetivo escalable, puede utilizar los siguientes ejemplos como ayuda para empezar.

Scale out

Para crear una política de escalado escalonado para ampliar (aumentar la capacidad)

1. Utilice el siguiente `cat` comando para almacenar una configuración de política de escalado escalonado en un archivo JSON denominado `config.json` en su directorio principal. El siguiente es un ejemplo de configuración con un tipo de ajuste `PercentChangeInCapacity` que aumenta la capacidad del objetivo escalable en función de los siguientes ajustes escalonados (suponiendo un umbral de CloudWatch alarma de 70):

- Aumente la capacidad en un 10 por ciento cuando el valor de la métrica sea superior o igual a 70 pero inferior a 85
- Aumente la capacidad en un 20 por ciento cuando el valor de la métrica sea superior o igual a 85 pero inferior a 95
- Aumente la capacidad en un 30 por ciento cuando el valor de la métrica sea mayor o igual a 95

```
$ cat ~/config.json
{
  "AdjustmentType": "PercentChangeInCapacity",
  "MetricAggregationType": "Average",
  "Cooldown": 60,
  "MinAdjustmentMagnitude": 1,
  "StepAdjustments": [
    {
      "MetricIntervalLowerBound": 0.0,
      "MetricIntervalUpperBound": 15.0,
      "ScalingAdjustment": 10
    },
    {
      "MetricIntervalLowerBound": 15.0,
      "MetricIntervalUpperBound": 25.0,
      "ScalingAdjustment": 20
    },
    {
      "MetricIntervalLowerBound": 25.0,
      "ScalingAdjustment": 30
    }
  ]
}
```

Para obtener más información, consulte la Referencia [StepScalingPolicyConfiguration](#) de la API Application Auto Scaling.

2. Utilice el siguiente [put-scaling-policy](#) comando, junto con el `config.json` archivo que creó, para crear una política de escalado denominada `amy-step-scaling-policy`.

Linux, macOS o Unix

```
aws application-autoscaling put-scaling-policy --service-namespace ecs \
```

```
--scalable-dimension ecs:service:DesiredCount \  
--resource-id service/my-cluster/my-service \  
--policy-name my-step-scaling-policy --policy-type StepScaling \  
--step-scaling-policy-configuration file://config.json
```

## Windows

```
aws application-autoscaling put-scaling-policy --service-namespace ecs ^  
--scalable-dimension ecs:service:DesiredCount ^  
--resource-id service/my-cluster/my-service ^  
--policy-name my-step-scaling-policy --policy-type StepScaling ^  
--step-scaling-policy-configuration file://config.json
```

## Salida

El resultado incluye el ARN que actúa como un nombre único para la política. Lo necesita para crear una CloudWatch alarma para su política. A continuación, se muestra un ejemplo del resultado.

```
{  
  "PolicyARN":  
    "arn:aws:autoscaling:region:123456789012:scalingPolicy:ac542982-cbeb-4294-891c-a5a941dfa787:resource/ecs/service/my-cluster/my-service:policyName/my-step-scaling-policy"  
}
```

## Scale in

Crear una política de escalado escalonado para ampliarlo (reducir la capacidad)

1. Utilice el siguiente `cat` comando para almacenar una configuración de política de escalado escalonado en un archivo JSON denominado `config.json` en su directorio principal. El siguiente es un ejemplo de configuración con un tipo de ajuste `ChangeInCapacity` que reduce la capacidad del objetivo escalable en función de los siguientes ajustes escalonados (suponiendo un umbral de CloudWatch alarma de 50):
  - Reduzca la capacidad en un 1% cuando el valor de la métrica sea inferior o igual a 50, pero superior a 40

- Reduzca la capacidad en 2 cuando el valor de la métrica sea inferior o igual a 40 pero superior a 30
- Disminuya la capacidad en 3 cuando el valor de la métrica sea inferior o igual a 30

```
$ cat ~/config.json
{
  "AdjustmentType": "ChangeInCapacity",
  "MetricAggregationType": "Average",
  "Cooldown": 60,
  "StepAdjustments": [
    {
      "MetricIntervalUpperBound": 0.0,
      "MetricIntervalLowerBound": -10.0,
      "ScalingAdjustment": -1
    },
    {
      "MetricIntervalUpperBound": -10.0,
      "MetricIntervalLowerBound": -20.0,
      "ScalingAdjustment": -2
    },
    {
      "MetricIntervalUpperBound": -20.0,
      "ScalingAdjustment": -3
    }
  ]
}
```

Para obtener más información, consulte la Referencia [StepScalingPolicyConfiguration](#) de la API Application Auto Scaling.

2. Utilice el siguiente [put-scaling-policy](#) comando, junto con el `config.json` archivo que creó, para crear una política de escalado denominada `my-step-scaling-policy`.

Linux, macOS o Unix

```
aws application-autoscaling put-scaling-policy --service-namespace ecs \
  --scalable-dimension ecs:service:DesiredCount \
  --resource-id service/my-cluster/my-service \
  --policy-name my-step-scaling-policy --policy-type StepScaling \
  --step-scaling-policy-configuration file://config.json
```

## Windows

```
aws application-autoscaling put-scaling-policy --service-namespace ecs ^
--scalable-dimension ecs:service:DesiredCount ^
--resource-id service/my-cluster/my-service ^
--policy-name my-step-scaling-policy --policy-type StepScaling ^
--step-scaling-policy-configuration file://config.json
```

## Salida

El resultado incluye el ARN que actúa como un nombre único para la política. Necesita este ARN para crear una CloudWatch alarma para su política. A continuación, se muestra un ejemplo del resultado.

```
{
  "PolicyARN":
  "arn:aws:autoscaling:region:123456789012:scalingPolicy:ac542982-cbeb-4294-891c-a5a941dfa787:resource/ecs/service/my-cluster/my-service:policyName/my-step-scaling-policy"
}
```

## Paso 3: Cree una alarma que invoque una política de escalado

Por último, utilice el siguiente CloudWatch [put-metric-alarm](#) comando para crear una alarma y utilizarla con su política de escalado escalonado. En este ejemplo, tiene una alarma basada en la utilización media de la CPU. La alarma está configurada para que adopte el estado ALARM si alcanza un umbral del 70 % durante al menos dos periodos de evaluación consecutivos de 60 segundos. Para especificar una CloudWatch métrica diferente o usar su propia métrica personalizada, especifique su nombre `--metric-name` y su espacio de nombres en `--namespace`

### Linux, macOS o Unix

```
aws cloudwatch put-metric-alarm --alarm-name Step-Scaling-AlarmHigh-ECS:service/my-cluster/my-service \
--metric-name CPUUtilization --namespace AWS/ECS --statistic Average \
--period 60 --evaluation-periods 2 --threshold 70 \
--comparison-operator GreaterThanOrEqualToThreshold \
--dimensions Name=ClusterName,Value=default Name=ServiceName,Value=sample-app-service \
```

```
--alarm-actions PolicyARN
```

## Windows

```
aws cloudwatch put-metric-alarm --alarm-name Step-Scaling-AlarmHigh-ECS:service/my-cluster/my-service ^
--metric-name CPUUtilization --namespace AWS/ECS --statistic Average ^
--period 60 --evaluation-periods 2 --threshold 70 ^
--comparison-operator GreaterThanOrEqualToThreshold ^
--dimensions Name=ClusterName,Value=default Name=ServiceName,Value=sample-app-service
^
--alarm-actions PolicyARN
```

## Describa las políticas de escalado por pasos para Application Auto Scaling mediante el AWS CLI

Puede describir todas las políticas de escalado de un espacio de nombres de servicio mediante el [describe-scaling-policies](#) comando. El siguiente ejemplo describe todas las políticas de escalado de todos los servicios de Amazon ECS. Para incluirlos solo para un servicio específico de Amazon ECS, añada la `--resource-id` opción.

```
aws application-autoscaling describe-scaling-policies --service-namespace ecs
```

Puede filtrar los resultados a solo las políticas de escalado por pasos mediante el parámetro `--query`. Para obtener más información acerca de syntax para query, consulte [Control de la salida de comandos de la AWS CLI](#) en la Guía del usuario de AWS Command Line Interface .

## Linux, macOS o Unix

```
aws application-autoscaling describe-scaling-policies --service-namespace ecs \
--query 'ScalingPolicies[?PolicyType==`StepScaling`]'
```

## Windows

```
aws application-autoscaling describe-scaling-policies --service-namespace ecs ^
--query "ScalingPolicies[?PolicyType==`StepScaling`]"
```

## Salida

A continuación, se muestra un ejemplo del resultado.

```
[
  {
    "PolicyARN": "PolicyARN",
    "StepScalingPolicyConfiguration": {
      "MetricAggregationType": "Average",
      "Cooldown": 60,
      "StepAdjustments": [
        {
          "MetricIntervalLowerBound": 0.0,
          "MetricIntervalUpperBound": 15.0,
          "ScalingAdjustment": 1
        },
        {
          "MetricIntervalLowerBound": 15.0,
          "MetricIntervalUpperBound": 25.0,
          "ScalingAdjustment": 2
        },
        {
          "MetricIntervalLowerBound": 25.0,
          "ScalingAdjustment": 3
        }
      ],
      "AdjustmentType": "ChangeInCapacity"
    },
    "PolicyType": "StepScaling",
    "ResourceId": "service/my-cluster/my-service",
    "ServiceNamespace": "ecs",
    "Alarms": [
      {
        "AlarmName": "Step-Scaling-AlarmHigh-ECS:service/my-cluster/my-
service",
        "AlarmARN": "arn:aws:cloudwatch:region:012345678910:alarm:Step-Scaling-
AlarmHigh-ECS:service/my-cluster/my-service"
      }
    ],
    "PolicyName": "my-step-scaling-policy",
    "ScalableDimension": "ecs:service:DesiredCount",
    "CreationTime": 1515024099.901
  }
]
```

# Elimine una política de escalado por pasos para Application Auto Scaling mediante el AWS CLI

Puede eliminar una política de escalado cuando ya no la necesite. Para eliminar tanto la política de escalado como la CloudWatch alarma asociada, complete las siguientes tareas.

Para eliminar una política de escalado

Utilice el comando [delete-scaling-policy](#).

Linux, macOS o Unix

```
aws application-autoscaling delete-scaling-policy --service-namespace ecs \  
--scalable-dimension ecs:service:DesiredCount \  
--resource-id service/my-cluster/my-service \  
--policy-name my-step-scaling-policy
```

Windows

```
aws application-autoscaling delete-scaling-policy --service-namespace ecs ^  
--scalable-dimension ecs:service:DesiredCount ^  
--resource-id service/my-cluster/my-service ^  
--policy-name my-step-scaling-policy
```

Para eliminar la CloudWatch alarma

Utilice el comando [delete-alarms](#). Puede eliminar una o más alarmas a la vez. Por ejemplo, utilice el siguiente comando para eliminar las alarmas Step-Scaling-AlarmHigh-ECS:service/my-cluster/my-service y Step-Scaling-AlarmLow-ECS:service/my-cluster/my-service.

```
aws cloudwatch delete-alarms --alarm-name Step-Scaling-AlarmHigh-ECS:service/my-  
cluster/my-service Step-Scaling-AlarmLow-ECS:service/my-cluster/my-service
```

# Tutorial: configuración del escalado automático para manejar una carga de trabajo pesada

En este tutorial, aprenderá a escalar horizontalmente y en función de las ventanas de tiempo en las que la aplicación tendrá una carga de trabajo más pesada que la normal. Esto es útil cuando se tiene una aplicación que de repente puede tener un gran número de visitantes en un horario regular o en una base estacional.

Puede utilizar una política de escalado de seguimiento de destino junto con escalado programado para gestionar la carga adicional. El escalado programado inicia automáticamente los cambios en su `MinCapacity` y `MaxCapacity` en su nombre, en función de una programación que especifique. Cuando una política de escalado de seguimiento de destino está activa en el recurso, puede escalar dinámicamente en función de la utilización actual de recursos, dentro del nuevo rango de capacidad mínimo y máximo.

Después de completar este tutorial, sabrá cómo:

- Utilizar el escalado programado para agregar capacidad adicional para cubrir una carga pesada antes de que llegue y, a continuación, eliminar la capacidad adicional cuando ya no sea necesaria.
- Utilice una política de escalado de seguimiento de destino para escalar la aplicación en función del uso de recurso actual.

## Contenido

- [Requisitos previos](#)
- [Paso 1: Registrar un onjetivo escalable](#)
- [Paso 2: Configuración de acciones programadas según sus requisitos](#)
- [Paso 3: Agregar una política de escalado de seguimiento de destino](#)
- [Paso 4: Siguietes pasos](#)
- [Paso 5: Eliminar](#)

## Requisitos previos

Este tutorial presupone que ya ha realizado las siguientes acciones:

- Creó un Cuenta de AWS.

- Instaló y configuró el AWS CLI.
- Se concedieron los permisos necesarios para registrar y anular el registro de los recursos como destinos escalables con Application Auto Scaling. Además, otorgó los permisos necesarios para crear políticas de escalado y acciones programadas. Para obtener más información, consulte [Identity and Access Management para Application Auto Scaling](#).
- Creó un recurso compatible en un entorno que no es de producción y está disponible para su uso en este tutorial. Si aún no cuenta con uno, créelo ahora. Para obtener información acerca de los servicios y los recursos de AWS que funcionan con Application Auto Scaling, consulte la sección [Servicios de AWS que puede usar con Application Auto Scaling](#).

#### Note

Mientras completa este tutorial, hay dos pasos en los que debe establecer los valores de capacidad mínima y máxima de su recurso en 0 para restablecer la capacidad actual en 0. Según el recurso que utilice con Application Auto Scaling, es posible que no pueda restablecer la capacidad actual en 0 a lo largo de estos pasos. Para ayudarle a solucionar el problema, un mensaje en el resultado indicará que la capacidad mínima no puede ser inferior al valor especificado y proporcionará el valor de capacidad mínima que el AWS recurso puede aceptar.

## Paso 1: Registrar un onjetivo escalable

Para empezar a registrar el recurso como un destino escalable con Auto Scaling de aplicaciones. Un destino escalable es un recurso que se puede escalar horizontalmente o reducir horizontalmente con Auto Scaling de aplicaciones.

Para registrar un objetivo escalable con Auto Scaling de aplicaciones

- Use el siguiente [register-scalable-target](#) comando para registrar un nuevo objetivo escalable. Establecimiento de los valores `--min-capacity` y `--max-capacity` a 0 para restablecer la capacidad actual a 0.

Reemplace el texto de muestra de `--service-namespace` por el espacio de nombres del servicio de AWS que está usando con Application Auto Scaling, `--scalable-dimension` por la dimensión escalable asociada al recurso que está registrando y `--resource-id` por un identificador del recurso. Estos valores varían en función de qué recurso se utiliza y cómo se

construye el ID del recurso. Consulte los temas en la sección [Servicios de AWS que puede usar con Application Auto Scaling](#) para obtener más información. Estos temas incluyen comandos de ejemplo que lo muestran cómo registrar objetivos escalables con Application Auto Scaling.

Linux, macOS o Unix

```
aws application-autoscaling register-scalable-target \  
  --service-namespace namespace \  
  --scalable-dimension dimension \  
  --resource-id identifier \  
  --min-capacity 0 --max-capacity 0
```

Windows

```
aws application-autoscaling register-scalable-target --service-namespace namespace \  
  --scalable-dimension dimension --resource-id identifier --min-capacity 0 --max-  
  capacity 0
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-  
id:scalable-target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

## Paso 2: Configuración de acciones programadas según sus requisitos

Puede usar el [put-scheduled-action](#) comando para crear acciones programadas que estén configuradas para satisfacer las necesidades de su empresa. En este tutorial, nos centramos en una configuración que deja de consumir recursos fuera de las horas de trabajo reduciendo la capacidad a 0.

Para crear una acción programada que se escale horizontalmente por la mañana

1. Para escalar el objetivo escalable, utilice el siguiente [put-scheduled-action](#) comando. Inclusión del parámetro `--schedule` con una programación periódica, en UTC, utilizando una expresión cron.

Según la programación especificada (todos los días a las 9:00 horas UTC), Auto Scaling de aplicaciones actualiza los valores MinCapacity y MaxCapacity al rango deseado de 1-5 unidades de capacidad.

Linux, macOS o Unix

```
aws application-autoscaling put-scheduled-action \  
  --service-namespace namespace \  
  --scalable-dimension dimension \  
  --resource-id identifier \  
  --scheduled-action-name my-first-scheduled-action \  
  --schedule "cron(0 9 * * ? *)" \  
  --scalable-target-action MinCapacity=1,MaxCapacity=5
```

Windows

```
aws application-autoscaling put-scheduled-action --service-namespace namespace --  
scalable-dimension dimension --resource-id identifier --scheduled-action-name my-  
first-scheduled-action --schedule "cron(0 9 * * ? *)" --scalable-target-action  
MinCapacity=1,MaxCapacity=5
```

Este comando no devuelve ningún resultado si se realiza correctamente.

2. Para confirmar que la acción programada existe, utilice el siguiente [describe-scheduled-actions](#) comando.

Linux, macOS o Unix

```
aws application-autoscaling describe-scheduled-actions \  
  --service-namespace namespace \  
  --query 'ScheduledActions[?ResourceId==`identifier`]'
```

Windows

```
aws application-autoscaling describe-scheduled-actions --service-  
namespace namespace --query "ScheduledActions[?ResourceId==`identifier`]"
```

A continuación, se muestra un ejemplo del resultado.

```
[
```

```

    {
      "ScheduledActionName": "my-first-scheduled-action",
      "ScheduledActionARN": "arn",
      "Schedule": "cron(0 9 * * ? *)",
      "ScalableTargetAction": {
        "MinCapacity": 1,
        "MaxCapacity": 5
      },
      ...
    }
  ]

```

Para crear una acción programada que se reduzca horizontalmente por la noche

1. Repita el procedimiento anterior para crear otra acción programada que Auto Scaling de aplicaciones utiliza para reducir horizontalmente al final del día.

Según la programación especificada (todos los días a las 20:00 UTC), Application Auto Scaling actualiza el MinCapacity Y del objetivo MaxCapacity a 0, según se indica en el siguiente [put-scheduled-action](#) comando.

Linux, macOS o Unix

```

aws application-autoscaling put-scheduled-action \
  --service-namespace namespace \
  --scalable-dimension dimension \
  --resource-id identifier \
  --scheduled-action-name my-second-scheduled-action \
  --schedule "cron(0 20 * * ? *)" \
  --scalable-target-action MinCapacity=0,MaxCapacity=0

```

Windows

```

aws application-autoscaling put-scheduled-action --service-namespace namespace --scalable-dimension dimension --resource-id identifier --scheduled-action-name my-second-scheduled-action --schedule "cron(0 20 * * ? *)" --scalable-target-action MinCapacity=0,MaxCapacity=0

```

2. Para confirmar que la acción programada existe, utilice el siguiente [describe-scheduled-actions](#) comando.

## Linux, macOS o Unix

```
aws application-autoscaling describe-scheduled-actions \  
  --service-namespace namespace \  
  --query 'ScheduledActions[?ResourceId==`identifier`]'
```

## Windows

```
aws application-autoscaling describe-scheduled-actions --service-  
namespace namespace --query "ScheduledActions[?ResourceId==`identifier`]"
```

A continuación, se muestra un ejemplo del resultado.

```
[  
  {  
    "ScheduledActionName": "my-first-scheduled-action",  
    "ScheduledActionARN": "arn",  
    "Schedule": "cron(0 9 * * ? *)",  
    "ScalableTargetAction": {  
      "MinCapacity": 1,  
      "MaxCapacity": 5  
    },  
    ...  
  },  
  {  
    "ScheduledActionName": "my-second-scheduled-action",  
    "ScheduledActionARN": "arn",  
    "Schedule": "cron(0 20 * * ? *)",  
    "ScalableTargetAction": {  
      "MinCapacity": 0,  
      "MaxCapacity": 0  
    },  
    ...  
  }  
]
```

## Paso 3: Agregar una política de escalado de seguimiento de destino

Ahora que dispone de la programación básica, agregue una política de escalado de seguimiento de destino para escalar en rol de la utilización actual de recursos.

Con el seguimiento de destino, Auto Scaling de aplicaciones compara el valor de destino de la política con el valor actual de la métrica especificada. Cuando son desiguales durante un periodo de tiempo, Auto Scaling de aplicaciones agrega o elimina capacidad para mantener un rendimiento constante. A medida que aumenta la carga en la aplicación y el valor de la métrica, Auto Scaling de aplicaciones agrega capacidad lo más rápido posible sin pasar por encima `MaxCapacity`. Cuando Auto Scaling de aplicaciones elimina la capacidad porque la carga es mínima, lo hace sin ir por debajo `MinCapacity`. Ahora que dispone de la programación básica, agregue una política de escalado de seguimiento de destino para escalar en función de la utilización actual de recursos.

Si la métrica no tiene datos suficientes porque la aplicación no tiene ninguna carga, Auto Scaling de aplicaciones no agrega ni elimina capacidad. En otras palabras, Application Auto Scaling prioriza la disponibilidad en situaciones en las que no hay suficiente información disponible.

Puede agregar varias políticas de escalado, pero asegúrese de no agregar políticas de escalado de pasos conflictivos, lo que podría provocar un comportamiento no deseado. Por ejemplo, si la política de escalado por pasos inicia una actividad de reducción horizontal antes de que la política de seguimiento de destino esté lista para la reducción horizontal, la actividad de reducción horizontal no se bloqueará. Una vez completada la actividad de reducir horizontalmente, la política de seguimiento de destino podría indicar a Auto Scaling de aplicaciones que vuelva a realizar el escalado horizontal.

Para crear una política de escalado de seguimiento de destino

1. Utilice el siguiente [put-scaling-policy](#) comando para crear la política.

Las métricas que se utilizan con más frecuencia para el seguimiento de objetivos están predefinidas y puede utilizarlas sin tener que proporcionar la especificación de métrica completa desde CloudWatch. Para obtener más información sobre las métricas predefinidas disponibles, consulte [Políticas de escalado de seguimiento de destino para Auto Scaling de aplicaciones](#).

Antes de ejecutar este comando, asegúrese de que la métrica predefinida espera el valor de destino. Por ejemplo, para escalar horizontalmente cuando la CPU alcanza el 50 % de uso, especifique un valor de destino de 50.0. O bien, para escalar horizontalmente la concurrencia aprovisionada de Lambda cuando el uso alcanza un 70 % de utilización, especifique un valor

de destino de 0.7. Para obtener información acerca de los valores de destino para un recurso concreto, consulte la documentación proporcionada por el servicio acerca de cómo configurar el seguimiento de destino. Para obtener más información, consulte [Servicios de AWS que puede usar con Application Auto Scaling](#).

Linux, macOS o Unix

```
aws application-autoscaling put-scaling-policy \
  --service-namespace namespace \
  --scalable-dimension dimension \
  --resource-id identifier \
  --policy-name my-scaling-policy --policy-type TargetTrackingScaling \
  --target-tracking-scaling-policy-configuration '{ "TargetValue": 50.0,
  "PredefinedMetricSpecification": { "PredefinedMetricType": "predefinedmetric" } }'
```

Windows

```
aws application-autoscaling put-scaling-policy --service-namespace namespace --
scalable-dimension dimension --resource-id identifier --policy-name my-scaling-
policy --policy-type TargetTrackingScaling --target-tracking-scaling-policy-
configuration "{ \"TargetValue\": 50.0, \"PredefinedMetricSpecification\":
{ \"PredefinedMetricType\": \"predefinedmetric\" } }"
```

Si se ejecuta correctamente, este comando devuelve los ARN y los nombres de las dos CloudWatch alarmas que se crearon en su nombre.

2. Para confirmar que la acción programada existe, utilice el siguiente [describe-scaling-policies](#) comando.

Linux, macOS o Unix

```
aws application-autoscaling describe-scaling-policies --service-namespace namespace
\
  --query 'ScalingPolicies[?ResourceId==`identifier`]'
```

Windows

```
aws application-autoscaling describe-scaling-policies --service-namespace namespace
--query "ScalingPolicies[?ResourceId==`identifier`]"
```

A continuación, se muestra un ejemplo del resultado.

```
[
  {
    "PolicyARN": "arn",
    "TargetTrackingScalingPolicyConfiguration": {
      "PredefinedMetricSpecification": {
        "PredefinedMetricType": "predefinedmetric"
      },
      "TargetValue": 50.0
    },
    "PolicyName": "my-scaling-policy",
    "PolicyType": "TargetTrackingScaling",
    "Alarms": [],
    ...
  }
]
```

## Paso 4: Sigüientes pasos

Cuando se produzca una actividad de escalado, verá un registro de ella en el resultado de las actividades de escalado para el objetivo escalable, por ejemplo:

```
Successfully set desired count to 1. Change successfully fulfilled by ecs.
```

Para monitorear sus actividades de escalado con Application Auto Scaling, puede usar el siguiente [describe-scaling-activities](#) comando.

Linux, macOS o Unix

```
aws application-autoscaling describe-scaling-activities
--service-namespace namespace \
--scalable-dimension dimension \
--resource-id identifier
```

Windows

```
aws application-autoscaling describe-scaling-activities --service-namespace namespace
--scalable-dimension dimension --resource-id identifier
```

## Paso 5: Eliminar

Para evitar que su cuenta acumule cargos por los recursos creados durante el escalado activo, puede limpiar la configuración de escalado asociada como se indica a continuación.

Al eliminar la configuración de escalado, no se elimina el AWS recurso subyacente. Tampoco lo devuelve a su capacidad original. Puede usar la consola del servicio donde creó el recurso para eliminarlo o ajustar su capacidad.

### Eliminar las acciones programadas

El siguiente [delete-scheduled-action](#) comando elimina una acción programada especificada. Tenga en cuenta que puede omitir este paso si desea mantener las acciones programadas para usarlas en el futuro.

Linux, macOS o Unix

```
aws application-autoscaling delete-scheduled-action \  
  --service-namespace namespace \  
  --scalable-dimension dimension \  
  --resource-id identifier \  
  --scheduled-action-name my-second-scheduled-action
```

Windows

```
aws application-autoscaling delete-scheduled-action --service-namespace namespace \  
  --scalable-dimension dimension --resource-id identifier --scheduled-action-name my-second-scheduled-action
```

Para eliminar la política de escalado

El siguiente [delete-scaling-policy](#) comando elimina una política de escalado de seguimiento de objetivos específica. Puede omitir este paso si desea mantener la política de escalado que creó.

Linux, macOS o Unix

```
aws application-autoscaling delete-scaling-policy \  
  --service-namespace namespace \  
  --scalable-dimension dimension \  
  --resource-id identifier \  
  --scaling-policy-name scaling-policy-name
```

```
--policy-name my-scaling-policy
```

## Windows

```
aws application-autoscaling delete-scaling-policy --service-namespace namespace --scalable-dimension dimension --resource-id identifier --policy-name my-scaling-policy
```

## Anular el registro del objetivo escalable

Usa el siguiente [deregister-scalable-target](#) comando para anular el registro del objetivo escalable. Si tiene alguna política de escalado creada o cualquier acción programada que todavía no se haya eliminado, se elimina con este comando. Tenga en cuenta que puede omitir este paso si desea mantener el destino escalable registrado para usarlo en el futuro.

## Linux, macOS o Unix

```
aws application-autoscaling deregister-scalable-target \  
--service-namespace namespace \  
--scalable-dimension dimension \  
--resource-id identifier
```

## Windows

```
aws application-autoscaling deregister-scalable-target --service-namespace namespace --scalable-dimension dimension --resource-id identifier
```

# Suspensión y reanudación del escalado para Application Auto Scaling

En este tema se explica cómo suspender y, a continuación, reanudar una o varias actividades de escalado para los destinos escalables de su aplicación. La característica suspender-reanudar se utiliza para poner en pausa temporalmente las actividades de escalado activadas por sus políticas de escalado y las acciones programadas. Esto puede resultar útil, por ejemplo, cuando no desea que el escalado automático interfiera mientras realiza un cambio o investiga un problema de configuración. Sus políticas de escalado y las acciones programadas se pueden conservar y cuando esté listo, se pueden reanudar las actividades de escalado.

En los comandos de CLI de ejemplo que aparece a continuación, pasa los parámetros con formato JSON en un archivo config.json. También puede pasar estos parámetros en la línea de comandos mediante comillas para entrecomillar la estructura de datos JSON. Para obtener más información, consulte [Uso de entrecomillado de cadenas en la AWS CLI](#) de la Guía del usuario de AWS Command Line Interface .

## Contenido

- [Actividades de escalado](#)
- [Suspender y reanudar las actividades de escalado](#)

### Note

Para obtener instrucciones sobre cómo suspender los procesos de escalado horizontal mientras las implementaciones de Amazon ECS están en curso, consulte la siguiente documentación:

[Implementaciones y escalado automáticos de servicios](#) en la Guía para desarrolladores de Amazon Elastic Container Service

## Actividades de escalado

Auto Scaling de aplicaciones permite poner las siguientes actividades de escalado en un estado suspendido:

- Todas las actividades de escalado descendente activadas por una política de escalado.

- Todas las actividades de escalado ascendente activadas por una política de escalado.
- Todas las actividades de escalado que implican acciones programadas.

Las siguientes descripciones explican lo que ocurre cuando se suspenden las distintas actividades de escalado. Todas ellas pueden suspenderse y reanudarse por separado. En función del motivo de la suspensión de una actividad de escalado, es posible que tenga que suspender varias actividades de escalado a la vez.

#### DynamicScalingInSuspended

- Auto Scaling de aplicaciones no elimina capacidad cuando se activa una política de escalado de seguimiento de destino o una política de escalado por pasos. Esto le permite deshabilitar temporalmente las actividades de escalado asociadas a las políticas de escalado sin eliminar las políticas de escalado ni las alarmas asociadas a ellas. CloudWatch Al reanudar la reducción horizontal, Auto Scaling de aplicaciones evalúa las políticas con umbrales de alarma que se han infringido.

#### DynamicScalingOutSuspended

- Auto Scaling de aplicaciones no agrega capacidad cuando se activa una política de escalado de seguimiento de destino o una política de escalado por pasos. Esto le permite deshabilitar temporalmente las actividades de escalamiento horizontal asociadas a las políticas de escalado sin eliminar las políticas de escalado ni sus alarmas asociadas. CloudWatch Al reanudar el escalado horizontal, Auto Scaling de aplicaciones evalúa las políticas con umbrales de alarma que se han infringido.

#### ScheduledScalingSuspended

- Auto Scaling de aplicaciones no inicia las acciones de escalado que están programadas para ejecutarse durante el periodo de suspensión. Cuando se reanuda el escalado programado, Auto Scaling de aplicaciones solo evalúa las acciones programadas cuyo tiempo de ejecución aún no ha transcurrido.

## Suspender y reanudar las actividades de escalado

Puede suspender y reanudar las distintas actividades de escalado o todas ellas para su destino escalable de Auto Scaling de aplicaciones.

### Note

Por cuestiones de brevedad, estos ejemplos ilustran cómo suspender y reanudar el escalado de una tabla de DynamoDB. Para especificar un destino escalable diferente, especifique su espacio de nombres en `--service-namespace`, su dimensión escalable en `--scalable-dimension` y su ID de recurso en `--resource-id`. Para obtener más información y ejemplos de cada servicio, consulte los temas de [Servicios de AWS que puede usar con Application Auto Scaling](#).

Para suspender una actividad de escalado

Abra una ventana de línea de comandos y utilice el comando [register-scalable-target](#) con la opción `--suspended-state` tal y como se indica a continuación.

Linux, macOS o Unix

```
aws application-autoscaling register-scalable-target --service-namespace dynamodb \  
  --scalable-dimension dynamodb:table:ReadCapacityUnits --resource-id table/my-table \  
  --suspended-state file://config.json
```

Windows

```
aws application-autoscaling register-scalable-target --service-namespace dynamodb --  
scalable-dimension dynamodb:table:ReadCapacityUnits --resource-id table/my-table --  
suspended-state file://config.json
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

Para suspender solo las actividades de escalado descendente que se activan mediante una política de escalado, especifique lo siguiente en config.json.

```
{
  "DynamicScalingInSuspended":true
}
```

Para suspender solo las actividades de escalado ascendente que se activan mediante una política de escalado, especifique lo siguiente en config.json.

```
{
  "DynamicScalingOutSuspended":true
}
```

Para suspender solo las actividades de escalado que implican acciones programadas, especifique lo siguiente en config.json.

```
{
  "ScheduledScalingSuspended":true
}
```

Para suspender todas las actividades de escalado

Utilice el comando [register-scalable-target](#) con la opción `--suspended-state` de la siguiente manera.

Linux, macOS o Unix

```
aws application-autoscaling register-scalable-target --service-namespace dynamodb \
  --scalable-dimension dynamodb:table:ReadCapacityUnits --resource-id table/my-table \
  --suspended-state file://config.json
```

Windows

```
aws application-autoscaling register-scalable-target --service-namespace dynamodb --
scalable-dimension dynamodb:table:ReadCapacityUnits --resource-id table/my-table --
suspended-state file://config.json
```

En este ejemplo se supone que el archivo config.json contiene los siguientes parámetros con formato JSON.

```
{
  "DynamicScalingInSuspended":true,
  "DynamicScalingOutSuspended":true,
  "ScheduledScalingSuspended":true
}
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-
target/1234abcd56ab78cd901ef1234567890ab123"
}
```

## Ver actividades de escalado suspendidas

Utilice el comando [describe-scalable-targets](#) para determinar qué actividades de escalado están en estado suspendido para un destino escalable.

Linux, macOS o Unix

```
aws application-autoscaling describe-scalable-targets --service-namespace dynamodb \
--scalable-dimension dynamodb:table:ReadCapacityUnits --resource-id table/my-table
```

Windows

```
aws application-autoscaling describe-scalable-targets --service-namespace dynamodb --
scalable-dimension dynamodb:table:ReadCapacityUnits --resource-id table/my-table
```

A continuación, se muestra un ejemplo del resultado.

```
{
  "ScalableTargets": [
    {
      "ServiceNamespace": "dynamodb",
      "ScalableDimension": "dynamodb:table:ReadCapacityUnits",
      "ResourceId": "table/my-table",
      "MinCapacity": 1,
      "MaxCapacity": 20,
      "SuspendedState": {
        "DynamicScalingOutSuspended": true,

```

```

        "DynamicScalingInSuspended": true,
        "ScheduledScalingSuspended": true
    },
    "CreationTime": 1558125758.957,
    "RoleARN": "arn:aws:iam::123456789012:role/aws-
service-role/dynamodb.application-autoscaling.amazonaws.com/
AWSServiceRoleForApplicationAutoScaling_DynamoDBTable"
    }
]
}

```

## Reanudar actividades de escalado

Cuando esté listo para reanudar la actividad de escalado, puede reanudarla utilizando el comando [register-scalable-target](#).

El siguiente comando de ejemplo reanuda todas las actividades de escalado para el destino escalable especificado.

Linux, macOS o Unix

```

aws application-autoscaling register-scalable-target --service-namespace dynamodb \
--scalable-dimension dynamodb:table:ReadCapacityUnits --resource-id table/my-table \
--suspended-state file://config.json

```

Windows

```

aws application-autoscaling register-scalable-target --service-namespace dynamodb --
scalable-dimension dynamodb:table:ReadCapacityUnits --resource-id table/my-table --
suspended-state file://config.json

```

En este ejemplo se supone que el archivo `config.json` contiene los siguientes parámetros con formato JSON.

```

{
  "DynamicScalingInSuspended":false,
  "DynamicScalingOutSuspended":false,
  "ScheduledScalingSuspended":false
}

```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

# Actividades de escalado para Application Auto Scaling

Application Auto Scaling supervisa CloudWatch las métricas de su política de escalado e inicia una actividad de escalado cuando se superan los umbrales. También inicia las actividades de escalado cuando se modifica el tamaño máximo o mínimo del destino escalable, ya sea de forma manual o al seguir un cronograma.

Cuando se produce una actividad de escalado, Application Auto Scaling realiza una de las siguientes acciones:

- Aumenta la capacidad del destino escalable (lo que se denomina escalado horizontal)
- Disminuye la capacidad del destino escalable (lo que se denomina reducción horizontal)

Puede consultar las actividades de escalado de las últimas seis semanas.

## Busque las actividades de escalado por objetivo escalable

Para ver las actividades de escalado de un destino escalable específico, utilice el siguiente comando [describe-scaling-activities](#).

Linux, macOS o Unix

```
aws application-autoscaling describe-scaling-activities --service-namespace ecs \  
  --scalable-dimension ecs:service:DesiredCount --resource-id service/my-cluster/my-  
service
```

Windows

```
aws application-autoscaling describe-scaling-activities --service-namespace ecs --  
scalable-dimension ecs:service:DesiredCount --resource-id service/my-cluster/my-service
```

A continuación, se muestra un ejemplo de respuesta, donde `Status Code` contiene el estado actual de la actividad y `Status Message` contiene información sobre el estado de la actividad de escalado.

```
{  
  "ScalingActivities": [  
    {  
      "ScalableDimension": "ecs:service:DesiredCount",
```

```
    "Description": "Setting desired count to 1.",
    "ResourceId": "service/my-cluster/my-service",
    "ActivityId": "e6c5f7d1-dbbb-4a3f-89b2-51f33e766399",
    "StartTime": 1462575838.171,
    "ServiceNamespace": "ecs",
    "EndTime": 1462575872.111,
    "Cause": "monitor alarm web-app-cpu-lt-25 in state ALARM triggered policy
web-app-cpu-lt-25",
    "StatusMessage": "Successfully set desired count to 1. Change successfully
fulfilled by ecs.",
    "StatusCode": "Successful"
  }
]
}
```

Para obtener una descripción de los campos de la respuesta, consulte la Referencia [ScalingActivity](#) de la API Application Auto Scaling.

Los siguientes códigos de estado indican cuándo el evento de escalado que lleva a la actividad de escalado alcanza un estado completo:

- **Successful**: el escalado se completó correctamente
- **Overridden**: la capacidad deseada se actualizó mediante un evento de escalado más reciente
- **Unfulfilled**: se agotó el tiempo de espera para escalar o el servicio de destino no puede cumplir con la solicitud
- **Failed**: falló el escalado con una excepción

#### Note

La actividad de escalado también puede tener un estado de `Pending` o `InProgress`. Todas las actividades de escalado tienen un estado `Pending` antes de que el servicio de destino responde. Una vez que el destino responde, el estado de la actividad de escalado cambia a `InProgress`.

## Incluya actividades no escaladas

De forma predeterminada, las actividades de escalado no reflejan los momentos en los que Application Auto Scaling toma la decisión de no escalar.

Por ejemplo, supongamos que un servicio de Amazon ECS supera el umbral máximo de una métrica determinada, pero el número de tareas ya está en el número máximo de tareas permitido. En este caso, Application Auto Scaling no escala horizontalmente el número de tareas deseado.

Para incluir las actividades que no están escaladas (actividades no escaladas) en la respuesta, agregue la opción `--include-not-scaled-activities` al comando [describe-scaling-activities](#).

Linux, macOS o Unix

```
aws application-autoscaling describe-scaling-activities --include-not-scaled-activities \
  --service-namespace ecs --scalable-dimension ecs:service:DesiredCount \
  --resource-id service/my-cluster/my-service
```

Windows

```
aws application-autoscaling describe-scaling-activities --include-not-scaled-activities \
  --service-namespace ecs --scalable-dimension ecs:service:DesiredCount --resource- \
  id service/my-cluster/my-service
```

#### Note

Si este comando arroja un error, asegúrese de haber actualizado la versión AWS CLI local a la última versión.

Para confirmar que la respuesta incluye las actividades no escaladas, el elemento `NotScaledReasons` se muestra en el resultado de algunas actividades de escalado fallidas, si no de todas.

```
{
  "ScalingActivities": [
    {
      "ScalableDimension": "ecs:service:DesiredCount",
      "Description": "Attempting to scale due to alarm triggered",
      "ResourceId": "service/my-cluster/my-service",
      "ActivityId": "4d759079-a31f-4d0c-8468-504c56e2eecf",
      "StartTime": 1664928867.915,
      "ServiceNamespace": "ecs",
      "Cause": "monitor alarm web-app-cpu-gt-75 in state ALARM triggered policy \
web-app-cpu-gt-75",
    }
  ]
}
```

```

        "StatusCode": "Failed",
        "NotScaledReasons": [
            {
                "Code": "AlreadyAtMaxCapacity",
                "MaxCapacity": 4
            }
        ]
    }
]
}

```

Para obtener una descripción de los campos de la respuesta, consulte la Referencia [ScalingActivity](#) de la API Application Auto Scaling.

Si se devuelve una actividad no escalada, según el código de motivo indicado en Code, atributos como CurrentCapacity, MaxCapacity y MinCapacity pueden estar presentes en la respuesta.

Para evitar grandes cantidades de entradas duplicadas, solo la primera actividad no escalada se registrará en el historial de actividades de escalado. Las actividades subsiguientes que no estén escaladas no generarán nuevas entradas, a menos que cambie el motivo por el que no se haya escalado.

## Códigos de motivo

Los siguientes son los códigos de motivos de una actividad no escalada.

Código de motivo	Definición			
AutoScalingAnticipatedFlapping	El algoritmo de escalado automático decidió no realizar ninguna acción de escalado porque esto provocaría fluctuaciones. La fluctuación es			

Código de motivo	Definición			
	<p>un bucle infinito de reducción horizontal y escalado horizontal. Es decir, si se realiza una acción de escalado, el valor de la métrica cambiaría para iniciar otra acción de escalado en la dirección inversa.</p>			
TargetServicePutResourceAsInscalable	<p>El servicio de destino ha colocado temporalmente el recurso en un estado no escalable. Application Auto Scaling volverá a intentarlo si se cumplen las condiciones de escalado automático o configuradas en la política de escalado.</p>			

Código de motivo	Definición			
AlreadyAtMaxCapacity	<p>La capacidad máxima que especificó bloquea el escalado.</p> <p>Si desea que Application Auto Scaling escale horizontalmente, debe aumentar la capacidad máxima.</p>			
AlreadyAtMinCapacity	<p>La capacidad mínima que especificó bloquea el escalado.</p> <p>Si desea que Application Auto Scaling reduzca horizontalmente, debe disminuir la capacidad mínima.</p>			
AlreadyAtDesiredCapacity	<p>El algoritmo de escalado automático calculó que la capacidad revisada era igual a la capacidad actual.</p>			

# Monitorización de aplicaciones: Auto Scaling

La supervisión es una parte importante del mantenimiento de la confiabilidad, la disponibilidad y el rendimiento de Application Auto Scaling y sus demás AWS soluciones. Debe recopilar los datos de supervisión de todas las partes de la AWS solución para poder depurar más fácilmente un error multipunto en caso de que se produzca. AWS proporciona herramientas de monitoreo para observar Application Auto Scaling, informar cuando algo anda mal y tomar medidas automáticas cuando sea apropiado.

Puede utilizar las siguientes funciones para ayudarle a administrar sus AWS recursos:

## AWS CloudTrail

Con AWS CloudTrail, puede rastrear las llamadas realizadas a la aplicación Auto Scaling API por o en su nombre Cuenta de AWS. CloudTrail almacena la información en archivos de registro en el bucket de Amazon S3 que especifique. También, puede identificar qué usuarios y cuentas llamaron a Application Auto Scaling, la dirección IP de origen de las llamadas y el momento en que se hicieron. Para obtener más información, consulte [Registre las API llamadas de Application Auto Scaling mediante AWS CloudTrail](#).

### Note

Para obtener información sobre otros AWS servicios que pueden ayudarle a registrar y recopilar datos sobre sus cargas de trabajo, consulte la [guía de registro y monitoreo para propietarios de aplicaciones](#) en la Guía AWS prescriptiva.

## Amazon CloudWatch

Amazon le CloudWatch ayuda a analizar los registros y, en tiempo real, a supervisar las métricas de sus AWS recursos y aplicaciones alojadas. Puede recopilar métricas y realizar un seguimiento de las métricas, crear paneles personalizados y definir alarmas que le advierten o que toman medidas cuando una métrica determinada alcanza el umbral que se especifique. Por ejemplo, puede CloudWatch hacer un seguimiento de la utilización de los recursos y notificarle cuando la utilización sea muy alta o cuando la alarma de la métrica se INSUFFICIENT\_DATA active. Para obtener más información, consulte [Supervise el uso de recursos escalables mediante CloudWatch](#).

CloudWatch también realiza un seguimiento AWS API de las métricas de uso de Application Auto Scaling. Puede usar estas métricas para configurar alarmas que le avisen cuando el volumen de API llamadas supere un umbral que usted defina. Para obtener más información, consulta [las estadísticas AWS de uso](#) en la Guía del CloudWatch usuario de Amazon.

## Amazon EventBridge

Amazon EventBridge es un servicio de bus de eventos sin servidor que facilita la conexión de sus aplicaciones con datos de diversas fuentes. EventBridge ofrece un flujo de datos en tiempo real desde sus propias aplicaciones, aplicaciones de software-as-a S-Service (SaaS) AWS y servicios, y dirige esos datos a destinos como Lambda. Esto le permite monitorear los eventos que ocurren en los servicios y crear arquitecturas basadas en eventos. Para obtener más información, consulte [Supervise los eventos de Application Auto Scaling con Amazon EventBridge](#).

## AWS Health Dashboard

El AWS Health Dashboard (PHD) muestra información y también proporciona notificaciones que se invocan cuando se producen cambios en el estado de los recursos. AWS La información se presenta de dos formas: en un panel donde se muestran los eventos recientes y próximos organizados por categorías, y en un registro de eventos que contiene todos los eventos de los últimos 90 días. Para obtener más información, consulte [Cómo empezar con su AWS Health Dashboard](#).

# Supervise el uso de recursos escalables mediante CloudWatch

Con Amazon CloudWatch, obtiene una visibilidad casi continua de sus aplicaciones en todos los recursos escalables. CloudWatch es un servicio de monitoreo de AWS recursos. Puede usarlo CloudWatch para recopilar métricas y realizar un seguimiento, configurar alarmas y reaccionar automáticamente ante los cambios en sus AWS recursos. También puede crear paneles para monitorear las métricas o los conjuntos de métricas específicos que necesita.

Cuando interactúa con los servicios que se integran con Application Auto Scaling, estos envían las métricas que se muestran en la siguiente tabla a CloudWatch. En CloudWatch, las métricas se agrupan primero por el espacio de nombres del servicio y, después, por las distintas combinaciones de dimensiones de cada espacio de nombres. Estas métricas pueden ayudarle a monitorear el uso de los recursos y planificar la capacidad de sus aplicaciones. Si la carga de trabajo de su aplicación no es constante, esto indica que debe considerar el uso del escalado automático. Para obtener descripciones detalladas de estas métricas, consulte la documentación de la métrica de interés.

## Contenido

- [CloudWatch métricas para monitorear el uso de los recursos](#)
- [Métricas predefinidas para políticas de escalado de seguimiento de destino](#)

## CloudWatch métricas para monitorear el uso de los recursos

En la siguiente tabla se enumeran las CloudWatch métricas que están disponibles para respaldar la supervisión del uso de los recursos. La lista no es exhaustiva, pero ofrece una idea general. Si no ve estas métricas en la CloudWatch consola, asegúrese de haber completado la configuración del recurso. Para obtener más información, consulta la [Guía del CloudWatch usuario de Amazon](#).

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
AppStream 2.0			
Flotas	AWS/ AppStream	Nombre: Available Capacity  Dimensión : flota	<a href="#">AppStream Métricas 2.0</a>
Flotas	AWS/ AppStream	Nombre: CapacityUtilization  Dimensión : flota	<a href="#">AppStream Métricas 2.0</a>
Aurora			
Réplicas	AWS/ RDS	Nombre: CPUUtilization	<a href="#">Métricas de nivel de clúster de Aurora</a>

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
		Dimensiones:DBClusterIdentifier, Role (READER)	
Réplicas	AWS/RDS	Nombre: DatabaseConnections  Dimensiones:DBClusterIdentifier, Role (READER)	<a href="#">Métricas de nivel de clúster de Aurora</a>
Amazon Comprehend			
Puntos de conexión de clasificación de documentos	AWS/Comprende	Nombre: InferenceUtilization  Dimensión: EndpointArn	<a href="#">Métricas de punto de conexión de Amazon Comprehend</a>

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Puntos de conexión del reconocedor de entidades	AWS/Comprende	Nombre: InferenceUtilization  Dimensión: EndpointArn	<a href="#">Métricas de punto de conexión de Amazon Comprehend</a>
DynamoDB			
Tablas e índices secundarios globales	AWS/DynamoDB	Nombre: ProvisionedReadCapacityUnits  Dimensiones: TableName, GlobalSecondaryIndexName	<a href="#">Métricas de DynamoDB</a>

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Tablas e índices secundarios globales	AWS/ DynamoDB	Nombre: ProvisionedWriteCapacityUnits  Dimensiones: TableName , GlobalSecondaryIndexName	<a href="#">Métricas de DynamoDB</a>
Tablas e índices secundarios globales	AWS/ DynamoDB	Nombre: ConsumedReadCapacityUnits  Dimensiones: TableName , GlobalSecondaryIndexName	<a href="#">Métricas de DynamoDB</a>

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Tablas e índices secundarios globales	AWS/ DynamoDB	Nombre: ConsumedWriteCapacityUnits  Dimensiones: TableName, GlobalSecondaryIndexName	<a href="#">Métricas de DynamoDB</a>
Amazon ECS			
Servicios	AWS/ ECS	Nombre: CPUUtilization  Dimensiones: ClusterName, ServiceName	<a href="#">ECS Métricas de Amazon</a>

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Servicios	AWS/ ECS	Nombre: MemoryUtilization  Dimensiones: ClusterName, ServiceName	<a href="#">ECSMétricas de Amazon</a>
Servicios	AWS/ Aplicación ELB	Nombre: RequestCountPerTarget  Dimensión: TargetGroup	<a href="#">Métricas del Equilibrador de carga de aplicación</a>
ElastiCache			
Clústeres (grupos de replicación)	AWS/ ElastiCache	Nombre: DatabaseMemoryUsageCountedForEvictPercentage  Dimensión: ReplicationGroupId	<a href="#">ElastiCache Métricas (RedisOSS)</a>

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Clústeres (grupos de replicación)	AWS/ ElastiCache	Nombre: DatabaseCapacityUsageCountedForEvictionPercentage  Dimensión: : ReplicationGroupId	<a href="#">ElastiCache Métricas (RedisOSS)</a>
Clústeres (grupos de replicación)	AWS/ ElastiCache	Nombre: EngineCPUUtilization  Dimensiones: ReplicationGroupId, Función (principal)	<a href="#">ElastiCache Métricas (RedisOSS)</a>

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Clústeres (grupos de replicación)	AWS/ ElastiCache	Nombre: EngineCPUUtilization  Dimensiones: ReplicationGroupId, Role (réplica)	<a href="#">ElastiCache Métricas (RedisOSS)</a>
Amazon EMR			
Clústeres	AWS/ ElasticMapReduce	Nombre: YARNMemoryAvailablePercentage  Dimensión: ClusterId	<a href="#">EMR Métricas de Amazon</a>
Amazon Keyspaces			

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Tablas	AWS/ Cassandra	Nombre: ProvisionedReadCapacityUnits  Dimensiones: Keyspace, TableName	<a href="#">Métricas de Amazon Keyspaces</a>
Tablas	AWS/ Cassandra	Nombre: ProvisionedWriteCapacityUnits  Dimensiones: Keyspace, TableName	<a href="#">Métricas de Amazon Keyspaces</a>
Tablas	AWS/ Cassandra	Nombre: ConsumedReadCapacityUnits  Dimensiones: Keyspace, TableName	<a href="#">Métricas de Amazon Keyspaces</a>

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Tablas	AWS/ Cassandra	Nombre: ConsumedWriteCapacityUnits  Dimensiones: Keyspace, TableName	<a href="#">Métricas de Amazon Keyspaces</a>
Lambda			
Simultaneidad aprovisionada	AWS/ Lambda	Nombre: ProvisionedConcurrencyUtilization  Dimensiones: FunctionName, Recurso	<a href="#">Métricas de función de Lambda</a>
Amazon MSK			

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Almacenamiento de agente	AWS/ Kafka	Nombre: KafkaData LogsDiskUsed  Dimensiones: Cluster Name	<a href="#">MSKMétricas de Amazon</a>
Almacenamiento de agente	AWS/ Kafka	Nombre: KafkaData LogsDiskUsed  Dimensiones: Cluster Name, Broker ID	<a href="#">MSKMétricas de Amazon</a>
Neptune			
Clústeres	AWS/ Neptuno	Nombre: CPUUtilization  Dimensiones:DBClusterIdentifier, Role (READER)	<a href="#">Métricas de Neptune</a>
SageMaker			

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Variantes de punto de conexión	AWS/SageMaker	Nombre: InvocationsPerInstance  Dimensiones: EndpointName, VariantName	<a href="#">Métricas de invocación</a>
Componentes de inferencias	AWS/SageMaker	Nombre: InvocationsPerCopy  Dimensiones: InferenceComponentName	<a href="#">Métricas de invocación</a>

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Simultaneidad aprovisionada para un punto de conexión sin servidor	AWS/SageMaker	Nombre: ServerlessProvisionedConcurrencyUtilization  Dimensiones: EndpointName, VariantName	<a href="#">Métricas de punto de conexión sin servidor</a>
Spot Fleet (AmazonEC2)			
Spot Fleets	AWS/EC2Spot	Nombre: CPUUtilization  Dimensión: FleetRequestId	<a href="#">Métricas de flota de spot</a>
Spot Fleets	AWS/EC2Spot	Nombre: NetworkIn  Dimensión: FleetRequestId	<a href="#">Métricas de flota de spot</a>

Recursos escalables	Espacio de nombres	CloudWatch métrica	Enlace a la documentación
Spot Fleets	AWS/EC2Spot	Nombre: NetworkOutput  Dimensión: : FleetRequestId	<a href="#">Métricas de flota de spot</a>
Spot Fleets	AWS/Aplicación ELB	Nombre: RequestCountPerTarget  Dimensión: : TargetGroup	<a href="#">Métricas del Equilibrador de carga de aplicación</a>

## Métricas predefinidas para políticas de escalado de seguimiento de destino

La siguiente tabla muestra los tipos de métricas predefinidos de [Application Auto Scaling API Reference](#) con su nombre de CloudWatch métrica correspondiente. Cada métrica predefinida representa una agregación de los valores de la CloudWatch métrica subyacente. El resultado es el uso promedio de los recursos durante un minuto, basado en un porcentaje, a menos que se indique lo contrario. Las métricas predefinidas solo se utilizan en el contexto de la configuración de políticas de escalado del seguimiento de destino.

Puede obtener más información sobre estas métricas en la documentación del servicio disponible en la tabla en [CloudWatch métricas para monitorear el uso de los recursos](#).

Tipo de métrica predefinido	CloudWatch nombre de la métrica
AppStream 2.0	
AppStreamAverageCapacityUtilization	CapacityUtilization
Aurora	
RDSReaderAverageCPUUtilization	CPUUtilization
RDSReaderAverageDatabaseConnections	DatabaseConnections <sup>1</sup>
Amazon Comprehend	
ComprehendInferenceUtilization	InferenceUtilization
DynamoDB	
DynamoDBReadCapacityUtilization	ProvisionedReadCapacityUnits, ConsumedReadCapacityUnits <sup>2</sup>
DynamoDBWriteCapacityUtilization	ProvisionedWriteCapacityUnits, ConsumedWriteCapacityUnits <sup>2</sup>
Amazon ECS	
ECSServiceAverageCPUUtilization	CPUUtilization
ECSServiceAverageMemoryUtilization	MemoryUtilization
ALBRequestCountPerTarget	RequestCountPerTarget <sup>1</sup>
ElastiCache	
ElastiCacheDatabaseMemoryUsageCountedForEvictPercentage	DatabaseMemoryUsageCountedForEvictPercentage

Tipo de métrica predefinido	CloudWatch nombre de la métrica
ElastiCacheDatabaseCapacityUsageCountedForEvictPercentage	DatabaseCapacityUsageCountedForEvictPercentage
ElastiCachePrimaryEngineCPUUtilization	EngineCPUUtilization
ElastiCacheReplicaEngineCPUUtilization	EngineCPUUtilization
Amazon Keyspaces	
CassandraReadCapacityUtilization	ProvisionedReadCapacityUnits, ConsumedReadCapacityUnits <sup>2</sup>
CassandraWriteCapacityUtilization	ProvisionedWriteCapacityUnits, ConsumedWriteCapacityUnits <sup>2</sup>
Lambda	
LambdaProvisionedConcurrencyUtilization	ProvisionedConcurrencyUtilization
Amazon MSK	
KafkaBrokerStorageUtilization	KafkaDataLogsDiskUsed
Neptune	
NeptuneReaderAverageCPUUtilization	CPUUtilization
SageMaker	
SageMakerVariantInvocationsPerInstance	InvocationsPerInstance <sup>1</sup>
SageMakerInferenceComponentInvocationsPerCopy	InvocationsPerCopy <sup>1</sup>

Tipo de métrica predefinido	CloudWatch nombre de la métrica
SageMakerVariantProvisionedConcurrencyUtilization	ServerlessProvisionedConcurrencyUtilization
SageMakerInferenceComponentConcurrentRequestsPerCopyHighResolution	ConcurrentRequestsPerCopy
SageMakerVariantConcurrentRequestsPerModelHighResolution	ConcurrentRequestsPerModel
Flota de spot	
EC2SpotFleetRequestAverageCPUUtilization	CPUUtilization <sup>3</sup>
EC2SpotFleetRequestAverageNetworkIn <sup>3</sup>	NetworkIn <sup>1 3</sup>
EC2SpotFleetRequestAverageNetworkOut <sup>3</sup>	NetworkOut <sup>1 3</sup>
ALBRequestCountPerTarget	RequestCountPerTarget <sup>1</sup>

<sup>1</sup> La métrica se basa en un recuento, no en un porcentaje.

<sup>2</sup> Para DynamoDB y Amazon Keyspaces, las métricas predefinidas son una agregación de dos métricas para permitir el escalado en función CloudWatch del consumo de rendimiento aprovisionado.

<sup>3</sup> Para obtener el mejor rendimiento de escalado, se debe utilizar la monitorización EC2 detallada de Amazon.

# Registre las API llamadas de Application Auto Scaling mediante AWS CloudTrail

Application Auto Scaling está integrado con AWS CloudTrail un servicio que proporciona un registro de las acciones realizadas por un usuario, un rol o una persona que Servicio de AWS utiliza la aplicación Auto ScalingAPI. CloudTrail captura todas las API llamadas de Application Auto Scaling como eventos. Las llamadas capturadas incluyen llamadas desde AWS Management Console y llamadas de código a la aplicación Auto ScalingAPI. Si crea una ruta, puede habilitar la entrega continua de CloudTrail eventos a un bucket de Amazon S3, incluidos los eventos para Application Auto Scaling. Si no configura una ruta, podrá ver los eventos más recientes en la CloudTrail consola, en el historial de eventos. Con la información recopilada por CloudTrail, puede determinar la solicitud que se realizó a Application Auto Scaling, la dirección IP desde la que se realizó la solicitud, quién la realizó, cuándo se realizó y detalles adicionales.

Para obtener más información CloudTrail, consulte la [Guía AWS CloudTrail del usuario](#).

## Información sobre Application Auto Scaling en CloudTrail

CloudTrail está habilitada Cuenta de AWS cuando crea la cuenta. Cuando se produce una actividad de Application Auto Scaling, esa actividad se registra en un CloudTrail evento junto con otros eventos de AWS servicio en el historial de eventos. Puede ver, buscar y descargar los eventos recientes en su Cuenta de AWS. Para obtener más información, consulte [Visualización de eventos con el historial de CloudTrail eventos](#).

Para obtener un registro continuo de los eventos en su entorno Cuenta de AWS, incluidos los eventos de Application Auto Scaling, cree una ruta. Un rastro permite CloudTrail entregar archivos de registro a un bucket de Amazon S3. De forma predeterminada, cuando se crea un registro de seguimiento en la consola, el registro de seguimiento se aplica a todas las Regiones de AWS. La ruta registra los eventos de todas las regiones de la AWS partición y envía los archivos de registro al bucket de Amazon S3 que especifique. Además, puede configurar otros Amazon Web Services para analizar más a fondo los datos de eventos recopilados en los CloudTrail registros y actuar en función de ellos. Para más información, consulte los siguientes temas:

- [Introducción a la creación de registros de seguimiento](#)
- [CloudTrail servicios e integraciones compatibles](#)
- [Configuración de SNS las notificaciones de Amazon para CloudTrail](#)

- [Recibir archivos de CloudTrail registro de varias regiones](#) y [recibir archivos de CloudTrail registro de varias cuentas](#)

Todas las acciones de Application Auto Scaling se registran CloudTrail y documentan en la [Application Auto Scaling API Reference](#). Por ejemplo, las llamadas a las `PutScalingPolicy` `DescribeScalingPolicies` acciones y las llamadas generan entradas en los archivos de CloudTrail registro. `DeleteScalingPolicy`

Cada entrada de registro o evento contiene información sobre quién generó la solicitud. La información de identidad del usuario lo ayuda a determinar lo siguiente:

- Si la solicitud se realizó con credenciales de usuario root o AWS Identity and Access Management (IAM).
- Si la solicitud se realizó con credenciales de seguridad temporales de un rol o fue un usuario federado.
- Si la solicitud la realizó otro AWS servicio.

Para obtener más información, consulte el [CloudTrail userIdentityelemento](#).

## Introducción a las entradas del archivo de registro de Application Auto Scaling

Un rastro es una configuración que permite la entrega de eventos como archivos de registro a un bucket de Amazon S3 que usted especifique. CloudTrail Los archivos de registro contienen una o más entradas de registro. Un evento representa una solicitud única de cualquier fuente e incluye información sobre la acción solicitada, la fecha y la hora de la acción, los parámetros de la solicitud, etc. CloudTrail Los archivos de registro no son un registro ordenado de las API llamadas públicas, por lo que no aparecen en ningún orden específico.

El siguiente ejemplo muestra una entrada de CloudTrail registro que demuestra la `DescribeScalableTargets` acción.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "Root",
    "principalId": "123456789012",
```

```
    "arn": "arn:aws:iam::123456789012:root",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2018-08-21T17:05:42Z"
      }
    }
  },
  "eventTime": "2018-08-16T23:20:32Z",
  "eventSource": "autoscaling.amazonaws.com",
  "eventName": "DescribeScalableTargets",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "72.21.196.68",
  "userAgent": "EC2 Spot Console",
  "requestParameters": {
    "serviceNamespace": "ec2",
    "scalableDimension": "ec2:spot-fleet-request:TargetCapacity",
    "resourceIds": [
      "spot-fleet-request/sfr-05ceaf79-3ba2-405d-e87b-612857f1357a"
    ]
  },
  "responseElements": null,
  "additionalEventData": {
    "service": "application-autoscaling"
  },
  "requestID": "0737e2ea-fb2d-11e3-bfd8-99133058e7bb",
  "eventID": "3fcfb182-98f8-4744-bd45-b38835ab61cb",
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012"
}
```

## Recursos relacionados

Con CloudWatch los registros, puede supervisar y recibir alertas sobre eventos específicos capturados por CloudTrail. Los eventos que se envían a los CloudWatch registros son aquellos configurados para que los registre su ruta, así que asegúrese de haber configurado su ruta o senderos para registrar los tipos de eventos que le interesa monitorear. CloudWatch Los registros pueden monitorear la información de los archivos de registro y notificarte cuando se alcanzan ciertos umbrales. También se pueden archivar los datos del registro en un almacenamiento de larga duración. Para obtener más información, consulte la [Guía del usuario de Amazon CloudWatch Logs](#) y

el tema [Supervisión de los archivos de CloudTrail registro con Amazon CloudWatch Logs](#) de la Guía del AWS CloudTrail usuario.

## Supervise los eventos de Application Auto Scaling con Amazon EventBridge

Amazon EventBridge, anteriormente denominado CloudWatch Events, le ayuda a supervisar los eventos que son específicos de Application Auto Scaling y a iniciar acciones específicas que utilizan otros Servicios de AWS. Los eventos de Servicios de AWS se envían EventBridge prácticamente en tiempo real.

Con EventBridge él, puede crear reglas que coincidan con los eventos entrantes y enviarlos a los objetivos para su procesamiento.

Para obtener más información, consulta [Cómo empezar a usar Amazon EventBridge](#) en la Guía del EventBridge usuario de Amazon.

### Eventos de Auto Scaling de aplicaciones

A continuación, se muestran los eventos de ejemplo de Application Auto Scaling. Los eventos se producen en la medida de lo posible.

Actualmente, solo los eventos que son específicos para escalarse al máximo y API las llamadas mediante CloudTrail están disponibles para Application Auto Scaling.

#### Tipos de eventos

- [Evento de cambio de estado: escalado al máximo](#)
- [Eventos para API llamadas a través de CloudTrail](#)

#### Evento de cambio de estado: escalado al máximo

El siguiente evento de ejemplo muestra que Application Auto Scaling aumentó (escaló horizontalmente) la capacidad del objetivo escalable hasta su límite de tamaño máximo. Si la demanda vuelve a aumentar, se impedirá que Application Auto Scaling escale el objetivo a un tamaño mayor porque ya está escalado a su tamaño máximo.

En el objeto `detail`, los valores para los atributos `resourceId`, `serviceNamespace` y `scalableDimension` identifican el destino escalable. Los valores de los atributos

`newDesiredCapacity` y `oldDesiredCapacity` se refieren a la nueva capacidad después del evento de escalamiento horizontal y a la capacidad original antes del evento de escalamiento horizontal. La `maxCapacity` es el límite de tamaño máximo del objetivo escalable.

```
{
  "version": "0",
  "id": "11112222-3333-4444-5555-666677778888",
  "detail-type": "Application Auto Scaling Scaling Activity State Change",
  "source": "aws.application-autoscaling",
  "account": "123456789012",
  "time": "2019-06-12T10:23:40Z",
  "region": "us-west-2",
  "resources": [],
  "detail": {
    "startTime": "2022-06-12T10:20:43Z",
    "endTime": "2022-06-12T10:23:40Z",
    "newDesiredCapacity": 8,
    "oldDesiredCapacity": 5,
    "minCapacity": 2,
    "maxCapacity": 8,
    "resourceId": "table/my-table",
    "scalableDimension": "dynamodb:table:WriteCapacityUnits",
    "serviceName": "dynamodb",
    "statusCode": "Successful",
    "scaledToMax": true,
    "direction": "scale-out"
  }
}
```

Para crear una regla que capture todos los eventos de cambio de estado `scaledToMax` para todos los destinos escalables, utilice el siguiente patrón de eventos de ejemplo.

```
{
  "source": [
    "aws.application-autoscaling"
  ],
  "detail-type": [
    "Application Auto Scaling Scaling Activity State Change"
  ],
  "detail": {
    "scaledToMax": [
      true
    ]
  }
}
```

```
}  
}
```

## Eventos para API llamadas a través de CloudTrail

Un rastro es una configuración que se AWS CloudTrail utiliza para entregar eventos como archivos de registro a un bucket de Amazon S3. CloudTrail los archivos de registro contienen entradas de registro. Un evento representa una entrada de registro e incluye información sobre la acción solicitada, la fecha y la hora de la acción o los parámetros de la solicitud. Para obtener información sobre cómo empezar CloudTrail, consulte [Creación de una ruta](#) en la Guía del AWS CloudTrail usuario.

Los eventos que se entregan a través de CloudTrail tienen `AWS API Call via CloudTrail` como `valordetail-type`.

El siguiente evento de ejemplo representa una entrada de archivo de CloudTrail registro que muestra que un usuario de la consola ejecutó la [RegisterScalableTarget](#) acción Application Auto Scaling.

```
{  
  "version": "0",  
  "id": "99998888-7777-6666-5555-444433332222",  
  "detail-type": "AWS API Call via CloudTrail",  
  "source": "aws.autoscaling",  
  "account": "123456789012",  
  "time": "2022-07-13T16:50:15Z",  
  "region": "us-west-2",  
  "resources": [],  
  "detail": {  
    "eventVersion": "1.08",  
    "userIdentity": {  
      "type": "IAMUser",  
      "principalId": "123456789012",  
      "arn": "arn:aws:iam::123456789012:user/Bob",  
      "accountId": "123456789012",  
      "accessKeyId": "AKIAIOSFODNN7EXAMPLE",  
      "sessionContext": {  
        "sessionIssuer": {  
          "type": "Role",  
          "principalId": "123456789012",  
          "arn": "arn:aws:iam::123456789012:role/Admin",  
          "accountId": "123456789012",  
          "userName": "Admin"  
        }  
      }  
    }  
  }  
}
```

```

    },
    "webIdFederationData": {},
    "attributes": {
      "creationDate": "2022-07-13T15:17:08Z",
      "mfaAuthenticated": "false"
    }
  }
},
"eventTime": "2022-07-13T16:50:15Z",
"eventSource": "autoscaling.amazonaws.com",
"eventName": "RegisterScalableTarget",
"awsRegion": "us-west-2",
"sourceIPAddress": "AWS Internal",
"userAgent": "EC2 Spot Console",
"requestParameters": {
  "resourceId": "spot-fleet-request/sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
  "serviceNamespace": "ec2",
  "scalableDimension": "ec2:spot-fleet-request:TargetCapacity",
  "minCapacity": 2,
  "maxCapacity": 10
},
"responseElements": null,
"additionalEventData": {
  "service": "application-autoscaling"
},
"requestID": "e9caf887-8d88-11e5-a331-3332aa445952",
"eventID": "49d14f36-6450-44a5-a501-b0fdcdfaeb98",
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "123456789012",
"eventCategory": "Management",
"sessionCredentialFromConsole": "true"
}
}

```

Para crear una regla basada en todos [DeleteScalingPolicy](#) y que incluya [DeregisterScalableTarget](#) API todos los destinos escalables, utilice el siguiente ejemplo de patrón de eventos:

```

{
  "source": [
    "aws.autoscaling"
  ]
}

```

```
],
"detail-type": [
  "AWS API Call via CloudTrail"
],
"detail": {
  "eventSource": [
    "autoscaling.amazonaws.com"
  ],
  "eventName": [
    "DeleteScalingPolicy",
    "DeregisterScalableTarget"
  ],
  "additionalEventData": {
    "service": [
      "application-autoscaling"
    ]
  }
}
```

Para obtener más información sobre el uso CloudTrail, consulte [Registre las API llamadas de Application Auto Scaling mediante AWS CloudTrail](#).



```
--scalable-dimension dimension \  
--resource-id identifier \  
--min-capacity 1 --max-capacity 10 \  
--tags environment=production,iscontainerbased=true
```

Si se ejecuta correctamente, este comando devolverá el ARN del destino escalable.

```
{  
  "ScalableTargetARN": "arn:aws:application-autoscaling:region:account-id:scalable-  
target/1234abcd56ab78cd901ef1234567890ab123"  
}
```

### Note

Si este comando arroja un error, asegúrese de haber actualizado la versión AWS CLI local a la última versión.

## Etiquetas para seguridad

Use etiquetas para comprobar que el solicitante (como un usuario o rol de IAM) tiene permisos para realizar determinadas acciones. Proporcione información de etiquetas en el elemento de condición de una política de IAM mediante una o más de las siguientes claves de condición:

- Utilice `aws:ResourceTag/tag-key: tag-value` para permitir (o denegar) acciones de los usuarios en destinos escalables con etiquetas específicas.
- Utilice `aws:RequestTag/tag-key: tag-value` para exigir que una etiqueta específica esté presente o no en una solicitud.
- Utilice `aws:TagKeys [tag-key, ...]` para exigir que las claves de etiqueta específicas estén presentes o no en una solicitud.

Por ejemplo, la siguiente política de IAM concede permisos al usuario para las acciones siguientes: `DeregisterScalableTarget`, `DeleteScalingPolicy` y `DeleteScheduledAction`. Sin embargo, también deniega la acción si el destino escalable sobre el que se actúa tiene la etiqueta **`environment=production`**.

```
{  
  "Version": "2012-10-17",
```

```

    "Statement": [
      {
        "Effect": "Allow",
        "Action": [
          "application-autoscaling:DeregisterScalableTarget",
          "application-autoscaling>DeleteScalingPolicy",
          "application-autoscaling>DeleteScheduledAction"
        ],
        "Resource": "*"
      },
      {
        "Effect": "Deny",
        "Action": [
          "application-autoscaling:DeregisterScalableTarget",
          "application-autoscaling>DeleteScalingPolicy",
          "application-autoscaling>DeleteScheduledAction"
        ],
        "Resource": "*",
        "Condition": {
          "StringEquals": {"aws:ResourceTag/environment": "production"}
        }
      }
    ]
  }
}

```

## Control del acceso a las etiquetas

Utilice las etiquetas para comprobar que el solicitante (como un rol o usuario de IAM) tiene permisos para agregar, modificar o eliminar etiquetas para destinos escalables.

Por ejemplo, puede crear una política de IAM que permita eliminar solo la etiqueta con la clave **temporary** de destinos escalables.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "application-autoscaling:UntagResource",
      "Resource": "*",
      "Condition": {

```

```
    "ForAllValues:StringEquals": { "aws:TagKeys": [temporary] }  
  }  
] }  
}
```

# Seguridad en Application Auto Scaling

La seguridad en la nube AWS es la máxima prioridad. Como AWS cliente, usted se beneficia de una arquitectura de centro de datos y red diseñada para cumplir con los requisitos de las organizaciones más sensibles a la seguridad.

La seguridad es una responsabilidad compartida entre usted AWS y usted. El [modelo de responsabilidad compartida](#) la describe como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta AWS los servicios en la AWS nube. AWS también le proporciona servicios que puede utilizar de forma segura. Los auditores externos prueban y verifican periódicamente la eficacia de nuestra seguridad como parte de los [AWS programas](#) de de . Para obtener más información sobre los programas de conformidad que se aplican a Application Auto Scaling, consulte [AWS los servicios incluidos en el ámbito de aplicación por programa de conformidad](#) y .
- Seguridad en la nube: su responsabilidad viene determinada por el AWS servicio que utilice. También es responsable de otros factores, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y los reglamentos vigentes.

Esta documentación le ayuda a comprender cómo aplicar el modelo de responsabilidad compartida cuando se utiliza Application Auto Scaling. En los siguientes temas, se le mostrará cómo configurar Application Auto Scaling para satisfacer sus objetivos de seguridad y conformidad. También aprenderá a usar otros AWS servicios que le ayudan a monitorear y proteger sus recursos de Application Auto Scaling.

## Contenido

- [Protección de datos en Application Auto Scaling](#)
- [Identity and Access Management para Application Auto Scaling](#)
- [Acceda a Application Auto Scaling mediante puntos VPC finales de interfaz](#)
- [Capacidad de Application Auto Scaling](#)
- [Seguridad de la infraestructura en Application Auto Scaling](#)
- [Validación de cumplimiento de Application Auto Scaling](#)

## Protección de datos en Application Auto Scaling

El [modelo de](#) se aplica a protección de datos en Application Auto Scaling. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global en la que se ejecutan todos los Nube de AWS. Usted es responsable de mantener el control sobre el contenido alojado en esta infraestructura. Usted también es responsable de las tareas de administración y configuración de seguridad para los Servicios de AWS que utiliza. Para obtener más información sobre la privacidad de los datos, consulte la sección [Privacidad de datos FAQ](#). Para obtener información sobre la protección de datos en Europa, consulte el [modelo de responsabilidad AWS compartida y](#) la entrada del GDPR blog sobre AWS seguridad.

Con fines de protección de datos, le recomendamos que proteja Cuenta de AWS las credenciales y configure los usuarios individuales con AWS IAM Identity Center o AWS Identity and Access Management (IAM). De esta manera, solo se otorgan a cada usuario los permisos necesarios para cumplir sus obligaciones laborales. También recomendamos proteger sus datos de la siguiente manera:

- Utilice la autenticación multifactorial (MFA) con cada cuenta.
- UseSSL/TLSpara comunicarse con AWS los recursos. Necesitamos TLS 1.2 y recomendamos TLS 1.3.
- Configure API y registre la actividad del usuario con AWS CloudTrail.
- Utilice soluciones de AWS cifrado, junto con todos los controles de seguridad predeterminados Servicios de AWS.
- Utilice servicios de seguridad administrados avanzados, como Amazon Macie, que lo ayuden a detectar y proteger los datos confidenciales almacenados en Amazon S3.
- Si necesita entre FIPS 140 y 3 módulos criptográficos validados para acceder a AWS través de una interfaz de línea de comandos o unaAPI, utilice un FIPS terminal. Para obtener más información sobre los FIPS puntos finales disponibles, consulte la [Norma federal de procesamiento de información \(\) FIPS 140-3](#).

Se recomienda encarecidamente no introducir nunca información confidencial o sensible, como, por ejemplo, direcciones de correo electrónico de clientes, en etiquetas o campos de formato libre, tales como el campo Nombre. Esto incluye cuando trabaja con Application Auto Scaling u otro Servicios de AWS mediante la consolaAPI, AWS CLI, o AWS SDKs. Cualquier dato que ingrese en etiquetas o campos de formato libre utilizados para nombres se puede emplear para los registros de facturación

o diagnóstico. Si proporciona una URL a un servidor externo, le recomendamos encarecidamente que no incluya información sobre las credenciales URL para validar su solicitud a ese servidor.

## Identity and Access Management para Application Auto Scaling

AWS Identity and Access Management (IAM) es un Servicio de AWS que ayuda al administrador a controlar de forma segura el acceso a AWS los recursos. IAM los administradores controlan quién puede autenticarse (iniciar sesión) y quién está autorizado (tiene permisos) para usar los recursos de Application Auto Scaling. IAM es un Servicio de AWS que puede utilizar sin coste adicional.

Para obtener IAM la documentación completa, consulte la [Guía IAM del usuario](#).

### Control de acceso

Aunque disponga de credenciales válidas para autenticar las solicitudes, si no tiene permisos, no podrá crear recursos de Application Auto Scaling ni obtener acceso a ellos. Por ejemplo, debe tener permisos para crear políticas de escalado, configurar el escalado programado, etc.

En las siguientes secciones se proporcionan detalles sobre cómo puede utilizar un IAM administrador IAM para proteger sus AWS recursos, controlando quién puede realizar API las acciones de Application Auto Scaling.

#### Contenido

- [Cómo funciona Application Auto Scaling con IAM](#)
- [AWS políticas gestionadas para Application Auto Scaling](#)
- [Roles vinculados a servicios para Application Auto Scaling](#)
- [Ejemplos de políticas basadas en identidad de Application Auto Scaling](#)
- [Solución de problemas de acceso a Application Auto Scaling](#)
- [Validación de permisos para las API llamadas de Application Auto Scaling a los recursos de destino](#)

### Cómo funciona Application Auto Scaling con IAM

#### Note

En diciembre de 2017, hubo una actualización de Application Auto Scaling que permitía utilizar varios roles vinculados a servicios con servicios integrados en Application Auto

Scaling. Se requieren IAM permisos específicos y un rol vinculado al servicio Application Auto Scaling (o un rol de servicio para el escalado EMR automático de Amazon) para que los usuarios puedan configurar el escalado.

Antes de administrar el acceso IAM a Application Auto Scaling, conozca qué IAM funciones están disponibles para usar con Application Auto Scaling.

IAM funciones que puede utilizar con Application Auto Scaling

IAM función	Compatibilidad de Application Auto Scaling
<a href="#">Políticas basadas en identidades</a>	Sí
<a href="#">Acciones de políticas</a>	Sí
<a href="#">Recursos de políticas</a>	Sí
<a href="#">Claves de condición de política (específicas del servicio)</a>	Sí
<a href="#">Políticas basadas en recursos</a>	No
<a href="#">ACLs</a>	No
<a href="#">ABAC(etiquetas en las políticas)</a>	Parcial
<a href="#">Credenciales temporales</a>	Sí
<a href="#">Roles de servicio</a>	Sí
<a href="#">Roles vinculados al servicio</a>	Sí

Para obtener una visión general de cómo Servicios de AWS funcionan Application Auto Scaling y otras funciones con la mayoría de IAM las funciones, consulte [Servicios de AWS esa función IAM](#) en la Guía del IAM usuario.

## Políticas Application Auto Scaling basadas en identidades

Compatibilidad con las políticas basadas en identidad: sí

Las políticas basadas en la identidad son documentos de política de JSON permisos que se pueden adjuntar a una identidad, como un IAM usuario, un grupo de usuarios o un rol. Estas políticas controlan qué acciones pueden realizar los usuarios y los roles, en qué recursos y en qué condiciones. Para obtener información sobre cómo crear una política basada en la identidad, consulte [Creación de IAM políticas](#) en la Guía del usuario. IAM

Con las políticas IAM basadas en la identidad, puede especificar las acciones y los recursos permitidos o denegados, así como las condiciones en las que se permiten o deniegan las acciones. No es posible especificar la entidad principal en una política basada en identidad porque se aplica al usuario o rol al que está adjunto. Para obtener más información sobre todos los elementos que puede utilizar en una JSON política, consulte la [referencia sobre los elementos de la IAM JSON política](#) en la Guía del IAM usuario.

### Ejemplos de políticas basadas en identidad de Application Auto Scaling

Para ver ejemplos de políticas basadas en identidad de Application Auto Scaling, consulte [Ejemplos de políticas basadas en identidad de Application Auto Scaling](#).

### Acciones

Compatibilidad con las acciones de política: sí

En una declaración de IAM política, puede especificar cualquier API acción de cualquier servicio compatible IAM. Para Application Auto Scaling, utilice el siguiente prefijo con el nombre de la API acción: `application-autoscaling:`. Por ejemplo, `application-autoscaling:RegisterScalableTarget`, `application-autoscaling:PutScalingPolicy` y `application-autoscaling:DeregisterScalableTarget`.

Para especificar varias acciones en una única instrucción, sepárelas con comas como se muestra en el siguiente ejemplo.

```
"Action": [  
    "application-autoscaling:DescribeScalingPolicies",  
    "application-autoscaling:DescribeScalingActivities"
```

Puede utilizar caracteres comodín para especificar varias acciones (\*). Por ejemplo, para especificar todas las acciones que comiencen con la palabra `Describe`, incluya la siguiente acción.

```
"Action": "application-autoscaling:Describe*"
```

Para obtener una lista de las acciones de Application Auto Scaling, consulte [Acciones definidas por AWS Application Auto Scaling](#) en la Referencia de autorización del servicio.

## Recursos

Compatibilidad con los recursos de políticas: sí

En una declaración IAM de política, el Resource elemento especifica el objeto o los objetos que cubre la declaración. En el caso de Application Auto Scaling, cada declaración de IAM política se aplica a los objetivos escalables que especifique mediante sus nombres de recursos de Amazon (ARNs).

El formato ARN de recursos para los objetivos escalables:

```
arn:aws:application-autoscaling:region:account-id:scalable-target/unique-identifier
```

Por ejemplo, puede indicar un objetivo escalable específico en su declaración de la ARN siguiente manera. El ID único (1234abcd56ab78cd901ef1234567890ab123) es un valor que Application Auto Scaling asigna al objetivo escalable.

```
"Resource": "arn:aws:application-autoscaling:us-east-1:123456789012:scalable-target/1234abcd56ab78cd901ef1234567890ab123"
```

También puede especificar todas las instancias que pertenecen a una cuenta específica si sustituye el identificador único por el carácter comodín (\*) del siguiente modo.

```
"Resource": "arn:aws:application-autoscaling:us-east-1:123456789012:scalable-target/*"
```

Para especificar todos los recursos, o si una API acción específica no es compatibleARNs, utilice un comodín (\*) como Resource elemento de la siguiente manera.

```
"Resource": "*"
```

Para obtener más información, consulte [Tipos de recursos definidos por AWS Application Auto Scaling](#) en la Referencia de autorización de servicios.

## Claves de condición

Compatibilidad con claves de condición de políticas específicas del servicio: sí

Puede especificar condiciones en las IAM políticas que controlan el acceso a los recursos de Application Auto Scaling. La declaración de política solo será efectiva si se cumplen las condiciones.

Application Auto Scaling admite las siguientes claves de condición definidas por el servicio que puede utilizar en políticas basadas en la identidad para determinar quién puede realizar las acciones de Application Auto Scaling. API

- `application-autoscaling:scalable-dimension`
- `application-autoscaling:service-namespace`

Para saber con qué API acciones de Application Auto Scaling puede utilizar una clave de condición, consulte [Acciones definidas por AWS Application Auto Scaling](#) en la Referencia de autorización del servicio. Para obtener más información sobre el uso de las claves de condición de Application Auto Scaling, consulte [Claves de condición de AWS Application Auto Scaling](#).

Para ver las claves de condición globales que están disponibles para todos los servicios, consulte [las claves de contexto de condición AWS globales](#) en la Guía del IAM usuario.

## Políticas basadas en recursos

Admite políticas basadas en recursos: no

Otros AWS servicios, como Amazon Simple Storage Service, admiten políticas de permisos basadas en recursos. Por ejemplo, puede asociar una política de permisos a un bucket de S3 para administrar los permisos de acceso a dicho bucket.

Application Auto Scaling no soporta políticas basadas en recursos.

## Listas de control de acceso ( ) ACLs

SoportesACLs: No

Application Auto Scaling no admite listas de control de acceso (ACLs).

## ABACcon Application Auto Scaling

Soportes ABAC (etiquetas en las políticas): parciales

El control de acceso basado en atributos (ABAC) es una estrategia de autorización que define los permisos en función de los atributos. En AWS, estos atributos se denominan etiquetas. Puede

adjuntar etiquetas a IAM entidades (usuarios o roles) y a muchos AWS recursos. Etiquetar entidades y recursos es el primer paso de ABAC. Luego, diseñe ABAC políticas para permitir las operaciones cuando la etiqueta del principal coincida con la etiqueta del recurso al que está intentando acceder.

ABAC es útil en entornos de rápido crecimiento y ayuda en situaciones en las que la administración de políticas se vuelve engorrosa.

Para controlar el acceso en función de etiquetas, debe proporcionar información de las etiquetas en el [elemento de condición](#) de una política utilizando las claves de condición `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`.

ABAC es posible para los recursos que admiten etiquetas, pero no todos admiten etiquetas. Las acciones programadas y las políticas de escalado no admiten etiquetas, pero los objetivos escalables sí. Para obtener más información, consulte [Compatibilidad de Application Auto Scaling con etiquetas](#).

Para obtener más información al respecto de ABAC, consulte [¿Qué es ABAC?](#) en la Guía IAM del usuario. Para ver un tutorial con los pasos de configuración de ABAC, consulte [Usar el control de acceso basado en atributos \(ABAC\)](#) en la Guía del IAM usuario.

## Uso de credenciales temporales con Application Auto Scaling

Compatibilidad con credenciales temporales: sí

Algunos Servicios de AWS no funcionan cuando inicias sesión con credenciales temporales. Para obtener información adicional, incluida la información sobre cuáles Servicios de AWS funcionan con credenciales temporales, consulta la sección [Servicios de AWS Cómo trabajar con credenciales temporales IAM](#) en la Guía del IAM usuario.

Está utilizando credenciales temporales si inicia sesión en ellas AWS Management Console mediante cualquier método excepto un nombre de usuario y una contraseña. Por ejemplo, cuando accedes a AWS mediante el enlace de inicio de sesión único (SSO) de tu empresa, ese proceso crea automáticamente credenciales temporales. También crea credenciales temporales de forma automática cuando inicia sesión en la consola como usuario y luego cambia de rol. Para obtener más información sobre el cambio de rol, consulte [Cambiar a un rol \(consola\)](#) en la Guía del IAM usuario.

Puede crear credenciales temporales manualmente con la tecla AWS CLI o AWS API. A continuación, puede utilizar esas credenciales temporales para acceder a AWS. AWS recomienda generar credenciales temporales de forma dinámica en lugar de utilizar claves de acceso a largo plazo. Para obtener más información, consulte [Credenciales de seguridad temporales en IAM](#).

## Roles de servicio

Compatibilidad con roles de servicio: sí

Si su EMR clúster de Amazon utiliza el escalado automático, esta función permite que Application Auto Scaling asuma una [función de servicio](#) en su nombre. Al igual que los roles vinculados a servicios, los roles de servicio permiten que el servicio acceda a los recursos de otros servicios para completar una acción en su nombre. Los roles de servicio aparecen en su IAM cuenta y son propiedad de la cuenta. Esto significa que un IAM administrador puede cambiar los permisos de este rol. Sin embargo, hacerlo podría deteriorar la funcionalidad del servicio.

Application Auto Scaling solo admite funciones de servicio para AmazonEMR. Para obtener documentación sobre la función de EMR servicio, consulta [Cómo usar el escalado automático con una política personalizada para grupos de instancias](#) en la Amazon EMR Management Guide.

### Note

Con la introducción de las funciones vinculadas al servicio, ya no se requieren varias funciones de servicio antiguas, por ejemplo, para Amazon ECS y Spot Fleet.

## Roles vinculados al servicio

Admite roles vinculados al servicio: sí

Un rol vinculado al servicio es un tipo de rol de servicio que está vinculado a un. Servicio de AWS El servicio puede asumir el rol para realizar una acción en su nombre. Los roles vinculados al servicio aparecen en usted Cuenta de AWS y son propiedad del servicio. Un IAM administrador puede ver los permisos de los roles vinculados al servicio, pero no editarlos.

Para obtener más información sobre los roles vinculados a servicios para Application Auto Scaling, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

## AWS políticas gestionadas para Application Auto Scaling

Una política AWS administrada es una política independiente creada y administrada por AWS. AWS Las políticas administradas están diseñadas para proporcionar permisos para muchos casos de uso comunes, de modo que pueda empezar a asignar permisos a usuarios, grupos y funciones.

Ten en cuenta que es posible que las políticas AWS administradas no otorguen permisos con privilegios mínimos para tus casos de uso específicos, ya que están disponibles para que los usen todos los AWS clientes. Se recomienda definir [políticas administradas por el cliente](#) específicas para sus casos de uso a fin de reducir aún más los permisos.

No puedes cambiar los permisos definidos en AWS las políticas administradas. Si AWS actualiza los permisos definidos en una política AWS administrada, la actualización afecta a todas las identidades principales (usuarios, grupos y roles) a las que está asociada la política. AWS es más probable que actualice una política AWS administrada cuando Servicio de AWS se lance una nueva o cuando haya nuevas API operaciones disponibles para los servicios existentes.

Para obtener más información, consulte [las políticas AWS administradas](#) en la Guía del IAM usuario.

## AWS política gestionada: AppStream 2.0 y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingAppStreamFleetPolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_AppStreamFleet](#) para permitir que Application Auto Scaling llame a Amazon AppStream CloudWatch y realice el escalado en su nombre.

### Detalles del permiso

La política de permisos permite a Application Auto Scaling realizar las siguientes acciones en todos los recursos relacionados («Recurso»: «\*»):

- Acción: `appstream:DescribeFleets`
- Acción: `appstream:UpdateFleet`
- Acción: `cloudwatch:DescribeAlarms`
- Acción: `cloudwatch:PutMetricAlarm`
- Acción: `cloudwatch>DeleteAlarms`

## AWS política gestionada: Aurora y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingRDSClusterPolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_RDSCluster](#) para permitir que Application Auto Scaling llame a Aurora CloudWatch y realice el escalado en su nombre.

## Detalles del permiso

La política de permisos permite a Application Auto Scaling realizar las siguientes acciones en todos los recursos relacionados («Recurso»: «\*»):

- Acción: `rds:AddTagsToResource`
- Acción: `rds:CreateDBInstance`
- Acción: `rds>DeleteDBInstance`
- Acción: `rds:DescribeDBClusters`
- Acción: `rds:DescribeDBInstance`
- Acción: `cloudwatch:DescribeAlarms`
- Acción: `cloudwatch:PutMetricAlarm`
- Acción: `cloudwatch>DeleteAlarms`

## AWS política gestionada: Amazon Comprehend y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingComprehendEndpointPolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_ComprehendEndpoint](#) para permitir que Application Auto Scaling llame a Amazon Comprehend CloudWatch y realice el escalado en su nombre.

## Detalles del permiso

La política de permisos permite a Application Auto Scaling realizar las siguientes acciones en todos los recursos relacionados («Recurso»: «\*»):

- Acción: `comprehend:UpdateEndpoint`
- Acción: `comprehend:DescribeEndpoint`
- Acción: `cloudwatch:DescribeAlarms`
- Acción: `cloudwatch:PutMetricAlarm`
- Acción: `cloudwatch>DeleteAlarms`

## AWS política gestionada: DynamoDB y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingDynamoDBTablePolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_DynamoDBTable](#) para permitir que Application Auto Scaling llame a DynamoDB y CloudWatch y realice el escalado en su nombre.

#### Detalles del permiso

La política de permisos permite a Application Auto Scaling realizar las siguientes acciones en todos los recursos relacionados («Recurso»: «\*»):

- Acción: dynamodb:DescribeTable
- Acción: dynamodb:UpdateTable
- Acción: cloudwatch:DescribeAlarms
- Acción: cloudwatch:PutMetricAlarm
- Acción: cloudwatch>DeleteAlarms

#### AWS política gestionada: Amazon ECS y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingECSServicePolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_ECSService](#) para permitir que Application Auto Scaling llame a Amazon ECS y CloudWatch y realice el escalado en su nombre.

#### Detalles del permiso

La política de permisos permite a Application Auto Scaling realizar las siguientes acciones en todos los recursos relacionados («Recurso»: «\*»):

- Acción: ecs:DescribeServices
- Acción: ecs:UpdateService
- Acción: cloudwatch:DescribeAlarms
- Acción: cloudwatch:PutMetricAlarm
- Acción: cloudwatch>DeleteAlarms

#### AWS política gestionada: ElastiCache y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingElastiCacheRGPpolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_ElastiCache RG](#) para permitir que Application Auto Scaling llame a ElastiCache CloudWatch y realice el escalado en su nombre.

#### Detalles del permiso

La política de permisos permite a Application Auto Scaling completar las siguientes acciones en los recursos especificados:

- Acción: `elasticache:DescribeReplicationGroups` en todos los recursos
- Acción: `elasticache:ModifyReplicationGroupShardConfiguration` en todos los recursos
- Acción: `elasticache:IncreaseReplicaCount` en todos los recursos
- Acción: `elasticache:DecreaseReplicaCount` en todos los recursos
- Acción: `elasticache:DescribeCacheClusters` en todos los recursos
- Acción: `elasticache:DescribeCacheParameters` en todos los recursos
- Acción: `cloudwatch:DescribeAlarms` en todos los recursos
- Acción: `cloudwatch:PutMetricAlarm` en el recurso `arn:aws:cloudwatch:*:*:alarm:TargetTracking*`
- Acción: `cloudwatch>DeleteAlarms` en el recurso `arn:aws:cloudwatch:*:*:alarm:TargetTracking*`
- Acción: `cloudwatch>DeleteAlarms`

#### AWS política gestionada: Amazon Keyspaces y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingCassandraTablePolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_CassandraTable](#) para permitir que Application Auto Scaling llame a Amazon Keyspaces CloudWatch y realice el escalado en su nombre.

#### Detalles del permiso

La política de permisos permite a Application Auto Scaling completar las siguientes acciones en los recursos especificados:

- Acción: `cassandra:Select` en los siguientes recursos:

- `arn:*:cassandra:*:*/keyspace/system/table/*`
- `arn:*:cassandra:*:*/keyspace/system_schema/table/*`
- `arn:*:cassandra:*:*/keyspace/system_schema_mcs/table/*`
- Acción: `cassandra:Alter` en todos los recursos
- Acción: `cloudwatch:DescribeAlarms` en todos los recursos
- Acción: `cloudwatch:PutMetricAlarm` en todos los recursos
- Acción: `cloudwatch>DeleteAlarms` en todos los recursos

## AWS política gestionada: Lambda y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingLambdaConcurrencyPolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_LambdaConcurrency](#) para permitir que Application Auto Scaling llame a Lambda CloudWatch y realice el escalado en su nombre.

### Detalles del permiso

La política de permisos permite a Application Auto Scaling realizar las siguientes acciones en todos los recursos relacionados («Recurso»: «\*»):

- Acción: `lambda:PutProvisionedConcurrencyConfig`
- Acción: `lambda:GetProvisionedConcurrencyConfig`
- Acción: `lambda>DeleteProvisionedConcurrencyConfig`
- Acción: `cloudwatch:DescribeAlarms`
- Acción: `cloudwatch:PutMetricAlarm`
- Acción: `cloudwatch>DeleteAlarms`

## AWS política gestionada: Amazon MSK y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingKafkaClusterPolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_KafkaCluster](#) para permitir que Application Auto Scaling llame a Amazon MSK CloudWatch y realice el escalado en su nombre.

## Detalles del permiso

La política de permisos permite a Application Auto Scaling realizar las siguientes acciones en todos los recursos relacionados («Recurso»: «\*»):

- Acción: `kafka:DescribeCluster`
- Acción: `kafka:DescribeClusterOperation`
- Acción: `kafka:UpdateBrokerStorage`
- Acción: `cloudwatch:DescribeAlarms`
- Acción: `cloudwatch:PutMetricAlarm`
- Acción: `cloudwatch>DeleteAlarms`

## AWS política gestionada: Neptune y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingNeptuneClusterPolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_NeptuneCluster](#) para permitir que Application Auto Scaling llame a Neptune CloudWatch y realice el escalado en su nombre.

## Detalles del permiso

La política de permisos permite a Application Auto Scaling completar las siguientes acciones en los recursos especificados:

- Acción: `rds:ListTagsForResource` en todos los recursos
- Acción: `rds:DescribeDBInstances` en todos los recursos
- Acción: `rds:DescribeDBClusters` en todos los recursos
- Acción: `rds:DescribeDBClusterParameters` en todos los recursos
- Acción: `cloudwatch:DescribeAlarms` en todos los recursos
- Acción: `rds:AddTagsToResource` en los recursos con el prefijo `autoscaled-reader` del motor de base de datos de Amazon Neptune (`"Condition": {"StringEquals": {"rds:DatabaseEngine": "neptune"}}`)
- Acción: `rds>CreateDBInstance` en los recursos con el prefijo `autoscaled-reader` de todos los clústeres de bases de datos (`"Resource": "arn:*:rds:*:*:db:autoscaled-reader*", "arn:aws:rds:*:*:cluster:*"`) del motor de base de datos de Amazon Neptune (`"Condition": {"StringEquals": {"rds:DatabaseEngine": "neptune"}}`)

- Acción: `rds>DeleteDBInstance` en el recurso `arn:aws:rds:*:*:db:autoscaled-reader*`
- Acción: `cloudwatch:PutMetricAlarm` en el recurso `arn:aws:cloudwatch:*:*:alarm:TargetTracking*`
- Acción: `cloudwatch>DeleteAlarms` en el recurso `arn:aws:cloudwatch:*:*:alarm:TargetTracking*`

## AWS política gestionada: SageMaker y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingSageMakerEndpointPolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_SageMakerEndpoint](#) para permitir que Application Auto Scaling llame SageMaker CloudWatch y realice el escalado en su nombre.

### Detalles del permiso

La política de permisos permite a Application Auto Scaling completar las siguientes acciones en los recursos especificados:

- Acción: `sagemaker:DescribeEndpoint` en todos los recursos
- Acción: `sagemaker:DescribeEndpointConfig` en todos los recursos
- Acción: `sagemaker:DescribeInferenceComponent` en todos los recursos
- Acción: `sagemaker:UpdateEndpointWeightsAndCapacities` en todos los recursos
- Acción: `sagemaker:UpdateInferenceComponentRuntimeConfig` en todos los recursos
- Acción: `cloudwatch:DescribeAlarms` en todos los recursos
- Acción: `cloudwatch:GetMetricData` en todos los recursos
- Acción: `cloudwatch:PutMetricAlarm` en el recurso `arn:aws:cloudwatch:*:*:alarm:TargetTracking*`
- Acción: `cloudwatch>DeleteAlarms` en el recurso `arn:aws:cloudwatch:*:*:alarm:TargetTracking*`

## AWS política gestionada: EC2 Spot Fleet y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingEC2SpotFleetRequestPolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_EC2SpotFleetRequest](#) para permitir que Application Auto Scaling llame a Amazon EC2 CloudWatch y realice el escalado en su nombre.

#### Detalles del permiso

La política de permisos permite a Application Auto Scaling realizar las siguientes acciones en todos los recursos relacionados («Recurso»: «\*»):

- Acción: `ec2:DescribeSpotFleetRequests`
- Acción: `ec2:ModifySpotFleetRequest`
- Acción: `cloudwatch:DescribeAlarms`
- Acción: `cloudwatch:PutMetricAlarm`
- Acción: `cloudwatch>DeleteAlarms`

#### AWS política gestionada: WorkSpaces y CloudWatch

Nombre de la política: [AWSApplicationAutoscalingWorkSpacesPoolPolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_WorkSpacesPool](#) para permitir que Application Auto Scaling llame WorkSpaces CloudWatch y realice el escalado en su nombre.

#### Detalles del permiso

La política de permisos permite a Application Auto Scaling completar las siguientes acciones en los recursos especificados:

- Acción: `workspaces:DescribeWorkspacesPools` en todos los recursos de la misma cuenta que SLR
- Acción: `workspaces:UpdateWorkspacesPool` en todos los recursos de la misma cuenta que SLR
- Acción: `cloudwatch:DescribeAlarms` en todas las alarmas de la misma cuenta que SLR
- Acción: `cloudwatch:PutMetricAlarm` en todas las alarmas de la misma cuenta que laSLR, donde el nombre de la alarma comience por TargetTracking
- Acción: `cloudwatch>DeleteAlarms` en todas las alarmas de la misma cuenta que laSLR, donde el nombre de la alarma comience por TargetTracking

## AWS política gestionada: recursos personalizados y CloudWatch

Nombre de la política: [AWSApplicationAutoScalingCustomResourcePolicy](#)

Esta política se adjunta a la función vinculada al servicio denominada [AWSServiceRoleForApplicationAutoScaling\\_CustomResource](#) para permitir que Application Auto Scaling llame a sus recursos personalizados que están disponibles a través de API Gateway CloudWatch y realice el escalado en su nombre.

### Detalles del permiso

La política de permisos permite a Application Auto Scaling realizar las siguientes acciones en todos los recursos relacionados («Recurso»: «\*»):

- Acción: `execute-api:Invoke`
- Acción: `cloudwatch:DescribeAlarms`
- Acción: `cloudwatch:PutMetricAlarm`
- Acción: `cloudwatch>DeleteAlarms`

## Application Auto Scaling actualiza las políticas AWS gestionadas

Vea los detalles sobre las actualizaciones de las políticas AWS administradas para Application Auto Scaling desde que este servicio comenzó a rastrear estos cambios. Para recibir alertas automáticas sobre los cambios en esta página, suscríbase al RSS feed de la página de historial del documento Application Auto Scaling.

Cambio	Descripción	Fecha
<a href="#">AWSApplicationAutoScalingWorkSpacesPoolPolicy</a> — Nueva política	Se ha añadido una política gestionada para Amazon WorkSpaces. Esta política está asociada a un <a href="#">rol vinculado a un servicio</a> que permite a Application Auto Scaling llamar WorkSpaces CloudWatch y realizar el escalado en su nombre.	24 de junio de 2024

Cambio	Descripción	Fecha
<p><a href="#">AWSApplicationAutoscalingSageMakerEndpointPolicy</a>: actualización de una política actual</p>	<p>Se agregaron permisos para llamar SageMaker DescribeInferenceComponent y UpdateInferenceComponentRuntimeConfig API acciones para respaldar la compatibilidad con el escalado automático de SageMaker los recursos para una próxima integración. La política ahora también restringe las CloudWatch alarmas CloudWatch PutMetricAlarm y las DeleteAlarms API acciones a las que se utilizan con las políticas de escalado y seguimiento de Target.</p>	<p>13 de noviembre de 2023</p>
<p><a href="#">AWSApplicationAutoscalingNeptuneClusterPolicy</a>— Nueva política</p>	<p>Se ha añadido una política gestionada para Neptune. Esta política está asociada a un <a href="#">rol vinculado a un servicio</a> que permite a Application Auto Scaling llamar a Neptune CloudWatch y realizar el escalado en su nombre.</p>	<p>6 de octubre de 2021</p>

Cambio	Descripción	Fecha
<a href="#">AWSApplicationAutoScalingRDSClusterPolicy</a> — Nueva política	Se ha añadido una política gestionada para ElastiCache. Esta política está asociada a un <a href="#">rol vinculado a un servicio</a> que permite a Application Auto Scaling llamar ElastiCache CloudWatch y realizar el escalado en su nombre.	19 de agosto de 2021
Application Auto Scaling comenzó a rastrear cambios	Application Auto Scaling comenzó a rastrear los cambios de sus políticas AWS administradas.	19 de agosto de 2021

## Roles vinculados a servicios para Application Auto Scaling

Application Auto Scaling utiliza [funciones vinculadas a servicios](#) para obtener los permisos que necesita para llamar a otros AWS servicios en su nombre. Un rol vinculado a un servicio es un tipo único de rol AWS Identity and Access Management (IAM) que está vinculado directamente a un servicio. AWS Los roles vinculados al servicio proporcionan una forma segura de delegar permisos a los AWS servicios, ya que solo el servicio vinculado puede asumir un rol vinculado al servicio.

Para los servicios que se integran con Application Auto Scaling, Application Auto Scaling crea roles vinculados a servicios para usted. Hay un rol vinculado a servicio para cada servicio. Cada rol vinculado a servicio confía en que la entidad de seguridad de servicio especificada lo asuma. Para obtener más información, consulte [Referencia de funciones vinculadas al servicio ARN](#).

Application Auto Scaling incluye todos los permisos necesarios para cada rol vinculado a un servicio. Estos permisos administrados se crean y administran mediante Application Auto Scaling y definen las acciones permitidas para cada tipo de recurso. Para obtener información detallada acerca de los permisos que cada rol concede, consulte [AWS políticas gestionadas para Application Auto Scaling](#).

### Contenido

- [Permisos necesarios para crear un rol vinculado a un servicio](#)
- [Creación de roles vinculados a servicios \(automático\)](#)

- [Creación de roles vinculados a servicios \(manual\)](#)
- [Edición de roles vinculados a servicios](#)
- [Eliminación de los roles vinculados a servicios](#)
- [Regiones admitidas de roles vinculados al servicio necesarios para Application Auto Scaling](#)
- [Referencia de funciones vinculadas al servicio ARN](#)

## Permisos necesarios para crear un rol vinculado a un servicio

Application Auto Scaling requiere permisos para crear un rol vinculado a un servicio la primera vez que un usuario de la empresa Cuenta de AWS llame a `RegisterScalableTarget` un servicio determinado. Application Auto Scaling crea un rol vinculado al servicio para el servicio de destino de su cuenta, si el rol aún no existe. El rol vinculado al servicio concede permisos a Application Auto Scaling para que pueda llamar al servicio de destino en su nombre.

Para que la creación automática de roles se realice correctamente, los usuarios deben disponer de permisos para la acción `iam:CreateServiceLinkedRole`.

```
"Action": "iam:CreateServiceLinkedRole"
```

La siguiente es una política basada en la identidad que concede permisos para crear un rol vinculado a un servicio para la flota de spot. Puede especificar el rol vinculado al servicio en el `Resource` campo de la política como una condición y el principal de servicio para su rol vinculado al servicio como condición ARN, como se muestra. Para cada servicio, ARN consulte. [Referencia de funciones vinculadas al servicio ARN](#)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:CreateServiceLinkedRole",
      "Resource": "arn:aws:iam::*:role/aws-service-role/ec2.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_EC2SpotFleetRequest",
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": "ec2.application-autoscaling.amazonaws.com"
        }
      }
    }
  ]
}
```

```

    }
  ]
}

```

### Note

La clave de `iam:AWSServiceName` IAM condición especifica el principal de servicio al que está asociado el rol, que se indica en este ejemplo de política como `ec2.application-autoscaling`.amazonaws.com. No trate de adivinar la entidad de seguridad de servicio. Para ver la entidad de seguridad de servicio de un servicio, consulte [Servicios de AWS que puede usar con Application Auto Scaling](#).

## Creación del roles vinculados a servicios (automático)

No necesita crear manualmente un rol vinculado a servicios. Application Auto Scaling; crea el rol vinculado al servicio correspondiente por usted cuando llama a `RegisterScalableTarget`. Por ejemplo, si configura el escalado automático para un ECS servicio de Amazon, Application Auto Scaling crea el `AWSServiceRoleForApplicationAutoScaling_ECSService` rol.

## Creación del roles vinculados a servicios (manual)

Para crear el rol vinculado al servicio, puede usar la IAM consola AWS CLI, o. IAM API Para obtener más información, consulte [Creación de un rol vinculado a un servicio](#) en la Guía del usuario. IAM

Para crear un rol vinculado a un servicio (AWS CLI)

Utilice el siguiente [create-service-linked-role](#) CLI comando para crear el rol vinculado al servicio Application Auto Scaling. En la solicitud, especifique el nombre del servicio “prefijo”.

Para buscar el prefijo de nombre de servicio, consulte la información acerca de la entidad de seguridad de servicio del rol vinculado al servicio para cada servicio en la sección [Servicios de AWS que puede usar con Application Auto Scaling](#). El nombre del servicio y la entidad de seguridad de servicio comparten el mismo prefijo. Por ejemplo, para crear el rol AWS Lambda vinculado al servicio, utilice. `lambda.application-autoscaling.amazonaws.com`

```
aws iam create-service-linked-role --aws-service-name prefix.application-autoscaling.amazonaws.com
```

## Edición de roles vinculados a servicios

En el caso de los roles vinculados a servicios creados por Application Auto Scaling, solo puede editar sus descripciones. Para obtener más información, consulte [Edición de un rol vinculado a un servicio](#) en la Guía del usuario. IAM

## Eliminación de los roles vinculados a servicios

Si ya no utiliza Application Auto Scaling con un servicio compatible, le recomendamos que elimine el rol vinculado al servicio correspondiente.

Solo puede eliminar un rol vinculado a un servicio después de eliminar los recursos de AWS relacionados. De este modo, evitará también que pueda revocar por accidente los permisos de Application Auto Scaling para los recursos. Para obtener más información, consulte la [documentación](#) del recurso escalable. Por ejemplo, para eliminar un ECS servicio de Amazon, consulta [Eliminar un servicio](#) en la Guía para desarrolladores de Amazon Elastic Container Service.

Se puede utilizar IAM para eliminar un rol vinculado a un servicio. Para obtener más información, consulte [Eliminar un rol vinculado a un servicio](#) en la Guía del usuario. IAM

Después de eliminar un rol vinculado a un servicio, Application Auto Scaling crea de nuevo el rol si se llama a RegisterScalableTarget.

## Regiones admitidas de roles vinculados al servicio necesarios para Application Auto Scaling

Application Auto Scaling admite el uso de funciones vinculadas al servicio en todas las AWS regiones en las que el servicio está disponible.

## Referencia de funciones vinculadas al servicio ARN

En la siguiente tabla se muestra el nombre de recurso de Amazon (ARN) del rol vinculado al servicio para cada uno de los Servicio de AWS que funcionan con Application Auto Scaling.

Servicio	ARN
AppStream 2.0	arn:aws:iam:: <i>012345678910</i> :role/aws-service-role/appstream.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_AppStreamFleet

Servicio	ARN
Aurora	arn:aws:iam:: <i>012345678910</i> :role/aws-service-role/rds.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_RDSCluster
Comprehend	arn:aws:iam:: <i>012345678910</i> :role/aws-service-role/comprehend.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_ComprehendEndpoint
DynamoDB	arn:aws:iam:: <i>012345678910</i> :role/aws-service-role/dynamodb.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_DynamoDBTable
ECS	arn:aws:iam:: <i>012345678910</i> :role/aws-service-role/ecs.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_ECSService
ElastiCache	arn:aws:iam:: <i>012345678910</i> :role/aws-service-role/elasticache.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_ElastiCacheRG
Keyspaces	arn:aws:iam:: <i>012345678910</i> :role/aws-service-role/cassandra.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_CassandraTable
Lambda	arn:aws:iam:: <i>012345678910</i> :role/aws-service-role/lambda.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_LambdaConcurrency
MSK	arn:aws:iam:: <i>012345678910</i> :role/aws-service-role/kafka.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_KafkaCluster

Servicio	ARN
Neptune	<code>arn:aws:iam:: 012345678910 :role/aws-service-role/neptune.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_NeptuneCluster</code>
SageMaker	<code>arn:aws:iam:: 012345678910 :role/aws-service-role/sagemaker.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_SageMakerEndpoint</code>
Spot Fleets	<code>arn:aws:iam:: 012345678910 :role/aws-service-role/ec2.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_EC2SpotFleetRequest</code>
WorkSpaces	<code>arn:aws:iam:: 012345678910 :role/aws-service-role/workspaces.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_WorkSpacesPool</code>
Recursos personalizados	<code>arn:aws:iam:: 012345678910 :role/aws-service-role/custom-resource.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_CustomResource</code>

### Note

Puedes especificar el ARN rol vinculado a un servicio para la RoleARN propiedad de un ScalableTarget recurso [AWS::ApplicationAutoScaling](#): en tus plantillas de AWS CloudFormation pila, incluso si el rol vinculado al servicio especificado aún no existe. Application Auto Scaling crea automáticamente el rol para usted.

## Ejemplos de políticas basadas en identidad de Application Auto Scaling

De forma predeterminada, un usuario nuevo no Cuenta de AWS tiene permisos para hacer nada. Un IAM administrador debe crear y asignar IAM políticas que otorguen permiso a una IAM identidad (como un usuario o un rol) para realizar API acciones de Application Auto Scaling.

Para obtener información sobre cómo crear una IAM política mediante los siguientes documentos de JSON política de ejemplo, consulte [Creación de políticas en la JSON pestaña](#) de la Guía del IAM usuario.

### Contenido

- [Permisos necesarios para las API acciones de Application Auto Scaling](#)
- [Los permisos necesarios para realizar API acciones en los servicios de destino y CloudWatch](#)
- [Permisos para trabajar en el AWS Management Console](#)

## Permisos necesarios para las API acciones de Application Auto Scaling

Las siguientes políticas otorgan permisos para casos de uso comunes al llamar a Application Auto ScalingAPI. Consulte esta sección cuando escriba políticas basadas en identidades. Cada política otorga permisos a todas o algunas de las API acciones de Application Auto Scaling. También debe asegurarse de que los usuarios finales tengan permisos para el servicio de destino y CloudWatch (consulte la siguiente sección para obtener más información).

La siguiente política basada en la identidad otorga permisos a todas las acciones de Application Auto ScalingAPI.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "application-autoscaling:*"
      ],
      "Resource": "*"
    }
  ]
}
```

La siguiente política basada en la identidad otorga permisos a todas las API acciones de Application Auto Scaling necesarias para configurar las políticas de escalado y no a las acciones programadas.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "application-autoscaling:RegisterScalableTarget",
        "application-autoscaling:DescribeScalableTargets",
        "application-autoscaling:DeregisterScalableTarget",
        "application-autoscaling:PutScalingPolicy",
        "application-autoscaling:DescribeScalingPolicies",
        "application-autoscaling:DescribeScalingActivities",
        "application-autoscaling>DeleteScalingPolicy"
      ],
      "Resource": "*"
    }
  ]
}
```

La siguiente política basada en la identidad otorga permisos a todas las API acciones de Application Auto Scaling necesarias para configurar las acciones programadas y no las políticas de escalado.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "application-autoscaling:RegisterScalableTarget",
        "application-autoscaling:DescribeScalableTargets",
        "application-autoscaling:DeregisterScalableTarget",
        "application-autoscaling:PutScheduledAction",
        "application-autoscaling:DescribeScheduledActions",
        "application-autoscaling:DescribeScalingActivities",
        "application-autoscaling>DeleteScheduledAction"
      ],
      "Resource": "*"
    }
  ]
}
```

```
}
```

## Los permisos necesarios para realizar API acciones en los servicios de destino y CloudWatch

Para configurar y utilizar correctamente Application Auto Scaling con el servicio de destino, los usuarios finales deben tener permisos para Amazon CloudWatch y para cada servicio de destino para el que vayan a configurar el escalado. Utilice las siguientes políticas para conceder los permisos mínimos necesarios para trabajar con los servicios de destino y CloudWatch.

### Contenido

- [AppStream Flotas 2.0](#)
- [Réplicas de Aurora](#)
- [Puntos de conexión de reconocedor de identidades y clasificación de documentos de Amazon Comprehend](#)
- [Tablas de DynamoDB e índices secundarios globales](#)
- [ECSservicios](#)
- [ElastiCache grupos de replicación](#)
- [EMRClústeres de Amazon](#)
- [Tablas de Amazon Keyspaces](#)
- [Funciones de Lambda](#)
- [Amazon Managed Streaming for Apache Kafka \(MSK\) broker Storage](#)
- [Clústeres de Neptune](#)
- [SageMaker puntos finales](#)
- [Flotas puntuales \(AmazonEC2\)](#)
- [Recursos personalizados](#)

### AppStream Flotas 2.0

La siguiente política basada en la identidad concede permisos a todos los AppStream 2.0 y a CloudWatch API las acciones necesarias.

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```
{
  "Effect": "Allow",
  "Action": [
    "appstream:DescribeFleets",
    "appstream:UpdateFleet",
    "cloudwatch:DescribeAlarms",
    "cloudwatch:PutMetricAlarm",
    "cloudwatch>DeleteAlarms"
  ],
  "Resource": "*"
}
]
```

## Réplicas de Aurora

La siguiente política basada en la identidad otorga permisos a todas las Aurora y a CloudWatch API las acciones que sean necesarias.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "rds:AddTagsToResource",
        "rds:CreateDBInstance",
        "rds>DeleteDBInstance",
        "rds:DescribeDBClusters",
        "rds:DescribeDBInstances",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
      ],
      "Resource": "*"
    }
  ]
}
```

## Puntos de conexión de reconocedor de identidades y clasificación de documentos de Amazon Comprehend

La siguiente política basada en la identidad otorga permisos a todas las acciones CloudWatch API y acciones de Amazon Comprehend que sean necesarias.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "comprehend:UpdateEndpoint",
        "comprehend:DescribeEndpoint",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
      ],
      "Resource": "*"
    }
  ]
}
```

## Tablas de DynamoDB e índices secundarios globales

La siguiente política basada en la identidad concede permisos a todas las acciones y DynamoDB que sean necesarias. CloudWatch API

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "dynamodb:DescribeTable",
        "dynamodb:UpdateTable",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
      ],
      "Resource": "*"
    }
  ]
}
```

```
}
```

## ECSservicios

La siguiente política basada en la identidad concede los permisos a todos los usuarios ECS y CloudWatch API las acciones necesarias.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ecs:DescribeServices",
        "ecs:UpdateService",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
      ],
      "Resource": "*"
    }
  ]
}
```

## ElastiCache grupos de replicación

La siguiente política basada en la identidad concede los permisos ElastiCache y las acciones necesarias a todas CloudWatch API las acciones necesarias.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticache:ModifyReplicationGroupShardConfiguration",
        "elasticache:IncreaseReplicaCount",
        "elasticache:DecreaseReplicaCount",
        "elasticache:DescribeReplicationGroups",
        "elasticache:DescribeCacheClusters",
        "elasticache:DescribeCacheParameters",
        "cloudwatch:DescribeAlarms",

```

```

        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
    ],
    "Resource": "*"
}
]
}

```

## EMR Clústeres de Amazon

La siguiente política basada en la identidad concede permisos a todos los Amazon EMR y CloudWatch API las acciones necesarias.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticmapreduce:ModifyInstanceGroups",
        "elasticmapreduce:ListInstanceGroups",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
      ],
      "Resource": "*"
    }
  ]
}

```

## Tablas de Amazon Keyspaces

La siguiente política basada en la identidad concede permisos a todos los Amazon Keyspaces y CloudWatch API las acciones que sean necesarios.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cassandra:Select",

```

```

        "cassandra:Alter",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
    ],
    "Resource": "*"
}
]
}

```

## Funciones de Lambda

La siguiente política basada en la identidad concede permisos a todas las CloudWatch API acciones y Lambda necesarias.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "lambda:PutProvisionedConcurrencyConfig",
        "lambda:GetProvisionedConcurrencyConfig",
        "lambda>DeleteProvisionedConcurrencyConfig",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
      ],
      "Resource": "*"
    }
  ]
}

```

## Amazon Managed Streaming for Apache Kafka (MSK) broker Storage

La siguiente política basada en la identidad concede permisos a todos los Amazon MSK y CloudWatch API las acciones necesarias.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",

```

```

    "Action": [
      "kafka:DescribeCluster",
      "kafka:DescribeClusterOperation",
      "kafka:UpdateBrokerStorage",
      "cloudwatch:DescribeAlarms",
      "cloudwatch:PutMetricAlarm",
      "cloudwatch>DeleteAlarms"
    ],
    "Resource": "*"
  }
]
}

```

## Clústeres de Neptune

La siguiente política basada en la identidad otorga permisos a todos los Neptune y CloudWatch API las acciones que sean necesarios.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "rds:AddTagsToResource",
        "rds:CreateDBInstance",
        "rds:DescribeDBInstances",
        "rds:DescribeDBClusters",
        "rds:DescribeDBClusterParameters",
        "rds>DeleteDBInstance",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
      ],
      "Resource": "*"
    }
  ]
}

```

## SageMaker puntos finales

La siguiente política basada en la identidad concede los permisos a todos los usuarios SageMaker y CloudWatch API las acciones necesarias.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sagemaker:DescribeEndpoint",
        "sagemaker:DescribeEndpointConfig",
        "sagemaker:DescribeInferenceComponent",
        "sagemaker:UpdateEndpointWeightsAndCapacities",
        "sagemaker:UpdateInferenceComponentRuntimeConfig",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
      ],
      "Resource": "*"
    }
  ]
}
```

## Flotas puntuales (AmazonEC2)

La siguiente política basada en la identidad otorga permisos a todas las flotas de Spot y a CloudWatch API las acciones necesarias.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeSpotFleetRequests",
        "ec2:ModifySpotFleetRequest",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
      ],
      "Resource": "*"
    }
  ]
}
```

## Recursos personalizados

La siguiente política basada en la identidad otorga permiso a Gateway para ejecutar la API acción. API Esta política también otorga permisos a todas CloudWatch las acciones necesarias.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "execute-api:Invoke",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch>DeleteAlarms"
      ],
      "Resource": "*"
    }
  ]
}
```

## Permisos para trabajar en el AWS Management Console

No existe una consola independiente para Application Auto Scaling. La mayoría de los servicios que se integran con Application Auto Scaling tienen características dedicadas que le ayudarán a configurar el escalado a través de su consola.

En la mayoría de los casos, cada servicio proporciona IAM políticas AWS administradas (predefinidas) que definen el acceso a su consola, lo que incluye permisos para las API acciones de Application Auto Scaling. Para obtener más información, consulte la documentación del servicio cuya consola desee utilizar.

También puede crear sus propias IAM políticas personalizadas para conceder a los usuarios permisos detallados para ver y trabajar con API acciones específicas de Application Auto Scaling en. AWS Management Console Puede utilizar las políticas de ejemplo de las secciones anteriores; sin embargo, están diseñadas para las solicitudes que se realizan con o una. AWS CLI SDK La consola utiliza API acciones adicionales para sus funciones, por lo que es posible que estas políticas no funcionen como se esperaba. Por ejemplo, para configurar el escalado por pasos, es posible que los usuarios necesiten permisos adicionales para crear y administrar CloudWatch las alarmas.

**Tip**

Para ayudarle a determinar qué API acciones son necesarias para realizar las tareas en la consola, puede utilizar un servicio como AWS CloudTrail. Para obtener más información, consulte la [AWS CloudTrail Guía del usuario de](#) .

La siguiente política basada en identidades concede permisos para configurar políticas de escalado para la flota de Spot. Además de los IAM permisos para Spot Fleet, el usuario de la consola que accede a la configuración de escalado de flotas desde la EC2 consola de Amazon debe tener los permisos adecuados para los servicios que admiten el escalado dinámico.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "application-autoscaling:*",
        "ec2:DescribeSpotFleetRequests",
        "ec2:ModifySpotFleetRequest",
        "cloudwatch:DeleteAlarms",
        "cloudwatch:DescribeAlarmHistory",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:DescribeAlarmsForMetric",
        "cloudwatch:GetMetricStatistics",
        "cloudwatch:ListMetrics",
        "cloudwatch:PutMetricAlarm",
        "cloudwatch:DisableAlarmActions",
        "cloudwatch:EnableAlarmActions",
        "sns:CreateTopic",
        "sns:Subscribe",
        "sns:Get*",
        "sns:List*"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iam:CreateServiceLinkedRole",
      "Resource": "arn:aws:iam::*:role/aws-service-role/ec2.application-autoscaling.amazonaws.com/AWSServiceRoleForApplicationAutoScaling_EC2SpotFleetRequest",
```

```
    "Condition": {
      "StringLike": {
        "iam:AWSServiceName": "ec2.application-autoscaling.amazonaws.com"
      }
    }
  ]
}
```

Esta política permite a los usuarios de la consola ver y modificar las políticas de escalado en la EC2 consola de Amazon y crear y gestionar CloudWatch alarmas en la CloudWatch consola.

Puede ajustar las API acciones para limitar el acceso de los usuarios. Por ejemplo, si sustituye `application-autoscaling:*` por `application-autoscaling:Describe*`, el usuario tendrá acceso de solo lectura.

También puede ajustar los CloudWatch permisos según sea necesario para limitar el acceso de los usuarios a CloudWatch las funciones. Para obtener más información, consulta [los permisos necesarios para la CloudWatch consola](#) en la Guía del CloudWatch usuario de Amazon.

## Solución de problemas de acceso a Application Auto Scaling

Si te encuentras con `AccessDeniedException` o dificultades similares al trabajar con Application Auto Scaling, consulte la información de esta sección.

### No tengo autorización para realizar una acción en Application Auto Scaling

Si recibe una `AccessDeniedException` al llamar a una AWS API operación, significa que las credenciales AWS Identity and Access Management (IAM) que está utilizando no tienen los permisos necesarios para realizar esa llamada.

En el siguiente ejemplo, el error se produce cuando el usuario de `mateojackson` intenta ver detalles sobre un destino escalable, pero no tiene permiso de `application-autoscaling:DescribeScalableTargets`.

```
An error occurred (AccessDeniedException) when calling the DescribeScalableTargets
operation: User: arn:aws:iam::123456789012:user/mateojackson is not authorized to
perform: application-autoscaling:DescribeScalableTargets
```

Si recibe este u otros errores similares, debe ponerse en contacto con su administrador para recibir ayuda.

El administrador de su cuenta deberá asegurarse de que tiene permisos para acceder a todas las API acciones que Application Auto Scaling utiliza para acceder a los recursos del servicio de destino y CloudWatch. Se requieren permisos diferentes en función de los recursos con los que se esté trabajando. Application Auto Scaling también requiere permiso para crear un rol vinculado a un servicio la primera vez que un usuario configura el escalado de un determinado recurso.

Soy administrador y mi IAM política arrojó un error o no funciona según lo esperado

Además de las acciones de Application Auto Scaling, sus IAM políticas deben conceder permisos para llamar al servicio de destino y CloudWatch. Si un usuario o una aplicación no tiene estos permisos adicionales, es posible que su acceso se deniegue inesperadamente. Para escribir IAM políticas para los usuarios y las aplicaciones en sus cuentas, consulte la información de [Ejemplos de políticas basadas en identidad de Application Auto Scaling](#).

Para obtener información acerca de cómo se lleva a cabo la validación, consulte [Validación de permisos para las API llamadas de Application Auto Scaling a los recursos de destino](#).

Tenga en cuenta que algunos problemas de permisos también pueden deberse a un problema con la creación de los roles vinculados al servicio utilizados por Application Auto Scaling. Para obtener más información sobre crear un rol vinculado al servicio, consulte [Roles vinculados a servicios para Application Auto Scaling](#).

## Validación de permisos para las API llamadas de Application Auto Scaling a los recursos de destino

Para realizar solicitudes autorizadas a las API acciones de Application Auto Scaling, es necesario que la API persona que llama tenga permisos para acceder a AWS los recursos del servicio de destino y dentro de él. CloudWatch Application Auto Scaling valida los permisos de las solicitudes asociadas al servicio de destino y CloudWatch antes de continuar con la solicitud. Para ello, realizamos una serie de llamadas para validar los IAM permisos de los recursos de destino. Cuando se devuelve una respuesta, la lee Application Auto Scaling. Si los IAM permisos no permiten una acción determinada, Application Auto Scaling no acepta la solicitud y devuelve un error al usuario con información sobre el permiso faltante. Esto garantiza que la configuración de escalado que el usuario desea implementar funcione según lo previsto y que se devuelva un error útil si se produce un error en la solicitud.

Como ejemplo de cómo funciona, la siguiente información proporciona detalles sobre cómo Application Auto Scaling realiza las validaciones de permisos con Aurora y. CloudWatch

Cuando un usuario llama a un `RegisterScalableTarget` API clúster de base de datos Aurora, `Application Auto Scaling` realiza todas las comprobaciones siguientes para comprobar que el usuario tiene los permisos necesarios (en negrita).

- `rds:c reateDBInstance`: Para determinar si el usuario tiene este permiso, enviamos una solicitud a la `CreateDBInstance` API operación para intentar crear una instancia de base de datos con parámetros no válidos (ID de instancia vacío) en el clúster de base de datos Aurora que especificó el usuario. En el caso de un usuario autorizado, API devuelve una respuesta en código de `InvalidParameterValue` error tras auditar la solicitud. Sin embargo, para un usuario no autorizado, obtenemos un error `AccessDenied` y falla la `Application Auto Scaling` aplicación con un error `ValidationException` al usuario que enumera los permisos que faltan.
- `rds:d eleteDBInstance`: Enviamos un ID de instancia vacío a la operación. `DeleteDBInstance` API Para un usuario autorizado, esta solicitud da como resultado un error `InvalidParameterValue`. Para un usuario no autorizado, resulta en `AccessDenied` y envía una excepción de validación al usuario (el mismo tratamiento que se describe en el primer punto).
- `rds:AddTagsToResource`: Como la `AddTagsToResource` API operación requiere un nombre de recurso de Amazon (ARN), es necesario especificar un recurso «ficticio» con un ID de cuenta no válido (12345) y un ID de instancia ficticio (non-existing-db) para construir el (). ARN `arn:aws:rds:us-east-1:12345:db:non-existing-db` Para un usuario autorizado, esta solicitud da como resultado un error `InvalidParameterValue`. Para un usuario no autorizado, resulta en `AccessDenied` y envía una excepción de validación al usuario.
- `rds:d escribeDBCluster`: Describimos el nombre del clúster del recurso que se está registrando para el escalado automático. Para un usuario autorizado, obtenemos un resultado de descripción válido. Para un usuario no autorizado, resulta en `AccessDenied` y envía una excepción de validación al usuario.
- `describeDBInstanceRDS:d`. Lo llamamos `DescribeDBInstance` API con un `db-cluster-id` filtro que filtra según el nombre del clúster que proporcionó el usuario para registrar el objetivo escalable. Para un usuario autorizado, podemos describir todas las instancias de base de datos en el clúster de base de datos. Para un usuario no autorizado, esta llamada da como resultado `AccessDenied` y envía una excepción de validación al usuario.
- `cloudwatch: PutMetricAlarm`: Lo llamamos `PutMetricAlarm` API sin ningún parámetro. Debido a que falta el nombre de la alarma, la solicitud da como resultado `ValidationError` para un usuario autorizado. Para un usuario no autorizado, resulta en `AccessDenied` y envía una excepción de validación al usuario.

- `cloudwatch:DescribeAlarms`: Llamamos al valor `DescribeAlarms` API con el número máximo de registros establecido en 1. Para un usuario autorizado, esperamos información sobre una alarma en la respuesta. Para un usuario no autorizado, esta llamada da como resultado `AccessDenied` y envía una excepción de validación al usuario.
- `cloudwatch:DeleteAlarms`: De manera similar a `PutMetricAlarm` lo anterior, no proporcionamos ningún parámetro que solicite `DeleteAlarms`. Debido a que falta un nombre de alarma en la solicitud, esta llamada no se produce un error con `ValidationError` para un usuario autorizado. Para un usuario no autorizado, resulta en `AccessDenied` y envía una excepción de validación al usuario.

Siempre que se produzca cualquiera de estos errores de validación, se registrará. Puede tomar medidas para identificar manualmente las llamadas que no superaron la validación mediante `AWS CloudTrail`. Para obtener más información, consulte la [AWS CloudTrail Guía del usuario de](#) .

#### Note

Si recibe alertas de eventos de `Application Auto Scaling` que estén utilizando `CloudTrail`, estas alertas incluirán las llamadas de `Application Auto Scaling` para validar los permisos de los usuarios de forma predeterminada. Para filtrar estas alertas, utilice el campo `invokedBy`, que contendrá `application-autoscaling.amazonaws.com` para estas comprobaciones de validación.

## Acceda a `Application Auto Scaling` mediante puntos VPC finales de interfaz

Puede usarlo `AWS PrivateLink` para crear una conexión privada entre su aplicación VPC y `Application Auto Scaling`. Puede acceder a `Application Auto Scaling` como si estuviera en su VPC casa, sin el uso de una puerta de enlace, NAT dispositivo, VPN conexión o `AWS Direct Connect` conexión de Internet. Las instancias VPC que tenga no necesitan direcciones IP públicas para acceder a `Application Auto Scaling`.

Esta conexión privada se establece mediante la creación de un punto de conexión de interfaz alimentado por `AWS PrivateLink`. Creamos una interfaz de red de punto de conexión en cada subred habilitada para el punto de conexión de interfaz. Se trata de interfaces de red administradas por el solicitante que sirven como punto de entrada para el tráfico destinado a `Application Auto Scaling`.

Para obtener más información, consulte [Acceso directo AWS PrivateLink en la Servicios de AWS guía](#).AWS PrivateLink

## Contenido

- [Cree un VPC punto final de interfaz](#)
- [Cree una política de puntos VPC finales](#)

## Cree un VPC punto final de interfaz

Cree un punto de conexión enlace para Application Auto Scaling utilizando el siguiente nombre de servicio:

```
com.amazonaws.region.application-autoscaling
```

Para obtener más información, consulte [Acceder a un AWS servicio mediante un VPC punto final de interfaz](#) en la AWS PrivateLink Guía.

No necesita cambiar cualquier otra configuración. Application Auto Scaling llama a otros AWS servicios mediante puntos finales de servicio o VPC puntos finales de interfaz privada, los que estén en uso.

## Cree una política de puntos VPC finales

Puede adjuntar una política a su VPC terminal para controlar el acceso a la aplicación Auto ScalingAPI. La política especifica:

- La entidad de seguridad que puede realizar acciones.
- Las acciones que se pueden realizar.
- El recurso en el que se pueden realizar las acciones.

El siguiente ejemplo muestra una política de VPC punto final que deniega a todos los usuarios el permiso para eliminar una política de escalado a través del punto final. La política de ejemplo también concede permiso a todos los usuarios para realizar todas las demás acciones.

```
{  
  "Statement": [  
    {
```

```
    "Action": "*",
    "Effect": "Allow",
    "Resource": "*",
    "Principal": "*"
  },
  {
    "Action": "application-autoscaling:DeleteScalingPolicy",
    "Effect": "Deny",
    "Resource": "*",
    "Principal": "*"
  }
]
```

Para obtener más información, consulte [las VPC políticas](#) de puntos finales en la AWS PrivateLink Guía.

## Capacidad de Application Auto Scaling

La infraestructura AWS global se basa en AWS regiones y zonas de disponibilidad.

AWS Las regiones proporcionan varias zonas de disponibilidad aisladas y separadas físicamente, que están conectadas mediante redes de baja latencia, alto rendimiento y alta redundancia.

Con las zonas de disponibilidad, puede diseñar y utilizar aplicaciones y bases de datos que realizan una conmutación por error automática entre las zonas sin interrupciones. Las zonas de disponibilidad tienen una mayor disponibilidad, tolerancia a errores y escalabilidad que las infraestructuras tradicionales de uno o varios centros de datos.

[Para obtener más información sobre AWS las regiones y las zonas de disponibilidad, consulte la infraestructura global.AWS](#)

## Seguridad de la infraestructura en Application Auto Scaling

Como servicio gestionado, Application Auto Scaling está protegido por la seguridad de la red AWS global. Para obtener información sobre los servicios AWS de seguridad y cómo se AWS protege la infraestructura, consulte [Seguridad AWS en la nube](#). Para diseñar su AWS entorno utilizando las mejores prácticas de seguridad de la infraestructura, consulte [Protección de infraestructuras en un marco](#) de buena AWS arquitectura basado en el pilar de la seguridad.

APIs llamadas AWS publicadas se utilizan para acceder a Application Auto Scaling a través de la red. Los clientes deben admitir lo siguiente:

- Seguridad de la capa de transporte (TLS). Necesitamos TLS 1.2 y recomendamos TLS 1.3.
- Cifre suites con perfecto secreto (PFS), como (Ephemeral Diffie-Hellman) o DHE ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La mayoría de los sistemas modernos como Java 7 y posteriores son compatibles con estos modos.

Además, las solicitudes deben firmarse con un identificador de clave de acceso y una clave de acceso secreta que esté asociada a un director. IAM También puede utilizar [AWS Security Token Service](#) (AWS STS) para generar credenciales de seguridad temporales para firmar solicitudes.

## Validación de cumplimiento de Application Auto Scaling

Para saber si uno Servicio de AWS está dentro del ámbito de aplicación de programas de cumplimiento específicos, consulte [Servicios de AWS Alcance por programa de cumplimiento Servicios de AWS](#) de cumplimiento y elija el programa de cumplimiento que le interese. Para obtener información general, consulte Programas de [AWS cumplimiento > Programas AWS](#) .

Puede descargar informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte [Descarga de informes en AWS Artifact](#) .

Su responsabilidad de cumplimiento al Servicios de AWS utilizarlos viene determinada por la confidencialidad de sus datos, los objetivos de cumplimiento de su empresa y las leyes y reglamentos aplicables. AWS proporciona los siguientes recursos para ayudar con el cumplimiento:

- [Guías de inicio rápido sobre seguridad y cumplimiento](#): estas guías de implementación analizan las consideraciones arquitectónicas y proporcionan los pasos para implementar entornos básicos centrados en AWS la seguridad y el cumplimiento.
- [Diseñando una arquitectura basada en la HIPAA seguridad y el cumplimiento en Amazon Web Services](#): en este documento técnico se describe cómo pueden utilizar las empresas AWS para crear HIPAA aplicaciones aptas.

### Note

No todos son aptos. Servicios de AWS HIPAA Para obtener más información, consulta la [Referencia de servicios HIPAA aptos](#).

- [AWS Recursos de cumplimiento](#): esta colección de libros de trabajo y guías puede aplicarse a su industria y ubicación.
- [AWS Guías de cumplimiento para clientes](#): comprenda el modelo de responsabilidad compartida desde el punto de vista del cumplimiento. En las guías se resumen las mejores prácticas para garantizar la seguridad Servicios de AWS y se orientan a los controles de seguridad en varios marcos (incluidos el Instituto Nacional de Estándares y Tecnología (NIST), el Consejo de Normas de Seguridad del Sector de las Tarjetas de Pago (PCI) y la Organización Internacional de Normalización (ISO)).
- [Evaluación de los recursos con reglas](#) en la guía para AWS Config desarrolladores: el AWS Config servicio evalúa en qué medida las configuraciones de los recursos cumplen con las prácticas internas, las directrices del sector y las normas.
- [AWS Security Hub](#)— Este Servicio de AWS proporciona una visión completa del estado de su seguridad interior AWS. Security Hub utiliza controles de seguridad para evaluar sus recursos de AWS y comprobar su cumplimiento con los estándares y las prácticas recomendadas del sector de la seguridad. Para obtener una lista de los servicios y controles compatibles, consulte la [Referencia de controles de Security Hub](#).
- [Amazon GuardDuty](#): Servicio de AWS detecta posibles amenazas para sus cargas de trabajo Cuentas de AWS, contenedores y datos mediante la supervisión de su entorno para detectar actividades sospechosas y maliciosas. GuardDuty puede ayudarlo a cumplir con varios requisitos de conformidad, por ejemplo PCIDSS, cumpliendo con los requisitos de detección de intrusiones exigidos por ciertos marcos de cumplimiento.
- [AWS Audit Manager](#)— Esto le Servicio de AWS ayuda a auditar continuamente su AWS consumo para simplificar la gestión del riesgo y el cumplimiento de las normativas y los estándares del sector.

## Cuotas para Auto Scaling de aplicaciones

Cuenta de AWS Tiene cuotas predeterminadas, anteriormente denominadas límites, para cada una de ellas Servicio de AWS. A menos que se indique lo contrario, cada cuota es específica de la región de . Puede solicitar el aumento de algunas cuotas, pero otras no se pueden aumentar.

Para ver las cuotas de Auto Scaling para aplicaciones, abra la [consola de Service Quotas](#). En el panel de navegación, elija AWS services y seleccione Application Auto Scaling.

Para solicitar un aumento de cuota, consulte [Solicitud de aumento de cuota](#) en la Guía del usuario de Service Quotas.

Cuenta de AWS Tiene las siguientes cuotas relacionadas con Application Auto Scaling.

Nombre	Valor predeterminado	Ajustable
Objetivos escalables por tipo de recurso	Amazon DynamoDB: 5000   Amazon ECS: 3000   Amazon Keyspaces: 1500   Otros tipos de recursos: 500	Sí
Políticas de escalado por objetivo escalable (tanto políticas de escalado escalado como de seguimiento de objetivos)	50	No
Acciones programadas por destino escalable	200	No
Ajustes de pasos por política de escalado de pasos	20	No

Tenga en cuenta las cuotas del servicio a medida que escale las cargas de trabajo. Por ejemplo, cuando alcance el número máximo de unidades de capacidad permitidas por un servicio, el escalado se detendrá. Si la demanda cae y la capacidad actual disminuye, Auto Scaling de aplicaciones puede realizar de nuevo el escalado horizontal. Para evitar volver a alcanzar este límite de capacidad, puede solicitar un aumento. Cada servicio tiene sus propias cuotas predeterminadas para la capacidad máxima del recurso. Para obtener información sobre las cuotas predeterminadas de otros servicios

---

de Amazon Web Services, consulte [Puntos de conexión y cuotas de servicios](#) en la Referencia general de Amazon Web Services.

# Historial de documentos de Application Auto Scaling

En la siguiente tabla se describen los cambios importantes de la documentación de Application Auto Scaling, a partir de enero de 2018. Para recibir notificaciones sobre las actualizaciones de esta documentación, puedes suscribirte al RSS feed.

Cambio	Descripción	Fecha
<a href="#">Agrega soporte para un grupo de WorkSpaces</a>	Utilice Application Auto Scaling para escalar un grupo de WorkSpaces. Para obtener más información, consulte <a href="#">Amazon WorkSpaces y Application Auto Scaling</a> . El tema <a href="#">Application Auto Scaling actualiza las políticas AWS administradas</a> para incluir una nueva política administrada con la que se puede integrar WorkSpaces.	27 de junio de 2024
<a href="#">Cambios en la guía</a>	Se ha actualizado la entrada Número máximo de destinos escalables por tipo de recurso en la documentación de cuotas. Consulte <a href="#">Cuotas para Application Auto Scaling</a> .	16 de enero de 2024
<a href="#">Support para componentes de SageMaker inferencia</a>	Utilice Application Auto Scaling para escalar copias de un componente de inferencia.	29 de noviembre de 2023
<a href="#">Actualización de los permisos de los IAM roles vinculados al servicio</a>	Application Auto Scaling actualiza la política de <code>AWSApplicationAutoScalingSageMakerEndpointPolicy</code> . Para	13 de noviembre de 2023

	<p>obtener más información, consulte <a href="#">Actualizaciones de Application Auto Scaling para políticas de AWS administradas</a>.</p>	
<a href="#">Support para la SageMaker simultaneidad aprovisionada sin servidor</a>	<p>Utilice Application Auto Scaling para escalar la simultaneidad aprovisionada de un punto de conexión sin servidor.</p>	9 de mayo de 2023
<a href="#">Clasifique sus destinos escalables mediante etiquetas</a>	<p>Ahora puede asignar metadatos a sus destinos escalables de Application Auto Scaling en forma de etiquetas . Consulte <a href="#">Compatibilidad con etiquetas para Application Auto Scaling</a>.</p>	20 de marzo de 2023
<a href="#">Support para matemáticas CloudWatch métricas</a>	<p>Ahora puede utilizar la calculadora de métricas al crear políticas de escalado de seguimiento de destino. Con la matemática métrica, puedes consultar múltiples CloudWatch métricas y usar expresiones matemáticas para crear nuevas series temporales basadas en estas métricas. Consulte <a href="#">Creación de una política de escalado de seguimiento de destino para Application Auto Scaling con métricas matemáticas</a>.</p>	14 de marzo de 2023

[Cambios en la guía](#)

Los nuevos temas de la Guía del usuario de Application Auto Scaling ayudan a empezar a usar AWS CloudShell Application Auto Scaling. Consulte [Utilización AWS CloudShell para trabajar con Application Auto Scaling desde la línea de comandos.](#)

17 de febrero de 2023

[Razones para no escalar](#)

Ahora puede recuperar los motivos legibles por máquina por los que Application Auto Scaling no escala sus recursos con Application Auto Scaling. API Consulte [Actividades de escalado para Application Auto Scaling.](#)

4 de enero de 2023

[Cambios en la guía](#)

Se ha actualizado la entrada Número máximo de destinos escalables por tipo de recurso en la documentación de cuotas. Consulte [Cuotas para Application Auto Scaling.](#)

6 de mayo de 2022

[Agregue compatibilidad con los clústeres de Amazon Neptune](#)

Utilice Application Auto Scaling para escalar el número de réplicas de un clúster de bases de datos de Amazon Neptune. Para obtener más información, consulte [Amazon Neptune y Application Auto Scaling](#). El tema [Actualizaciones de Application Auto Scaling de las políticas administradas de AWS](#) se ha actualizado para que incluya una nueva política administrada para la integración con Neptune.

6 de octubre de 2021

[Application Auto Scaling ahora informa de los cambios en sus políticas AWS gestionadas](#)

A partir del 19 de agosto de 2021, los cambios en las políticas administradas se informan en el tema [Application Auto Scaling updates to AWS managed policies](#). El primer cambio de la lista es la adición de los permisos necesarios para ElastiCache (RedisOSS).

19 de agosto de 2021

[Agregue soporte para grupos de ElastiCache replicación \(RedisOSS\)](#)

Utilice Application Auto Scaling para escalar el número de grupos de nodos y el número de réplicas por grupo de nodos para un grupo de replicación ElastiCache (clústerOSS) (Redis). Para obtener más información, consulte [ElastiCache \(RedisOSS\) y Application Auto Scaling](#).

19 de agosto de 2021

[Cambios en la guía](#)

IAM Los nuevos temas de la Guía del usuario de Application Auto Scaling le ayudan a solucionar problemas de acceso a Application Auto Scaling. Para obtener más información, consulte [Identity and Access Management para Application Auto Scaling](#). También se agregaron nuevos ejemplos de políticas de IAM permisos para acciones en los servicios de destino y Amazon CloudWatch. Para obtener más información, consulta [Ejemplos de políticas para trabajar con AWS CLI o un SDK](#).

23 de febrero de 2021

[Agregue compatibilidad con las zonas horarias locales](#)

Ahora puede crear acciones programadas en la zona horaria local. Si tu zona horaria sigue el horario de verano, se ajustará automáticamente al horario de verano (DST). Para obtener más información, consulte [Escalado programado](#).

2 de febrero de 2021

[Cambios en la guía](#)

Un nuevo [tutorial](#) en la Guía del usuario de Application Auto Scaling le ayuda a comprender cómo utilizar las políticas de escalado de seguimiento de destino y el escalado programado para aumentar la disponibilidad de la aplicación al utilizar Application Auto Scaling. Además, en un [tema](#) nuevo se explica cómo activar una notificación cuando CloudWatch se detecta algún problema que pueda requerir su atención.

15 de octubre de 2020

[Agregue compatibilidad con el almacenamiento de clúster de Amazon Managed Streaming for Apache Kafka](#)

Usa una política de escalado de seguimiento objetivo para ampliar la cantidad de almacenamiento de los agentes asociada a un MSK clúster de Amazon.

30 de septiembre de 2020

---

<a href="#">Agregue compatibilidad con los puntos de conexión del reconocedor de entidades de Amazon Comprehend</a>	Utilice Application Auto Scaling para escalar el número de unidades de inferencia aprovisionadas para los puntos de conexión del reconocedor de entidades de Amazon Comprehend.	28 de septiembre de 2020
<a href="#">Agregar compatibilidad con las tablas de Amazon Keyspaces (for Apache Cassandra)</a>	Utilice Application Auto Scaling para escalar el rendimiento aprovisionado (capacidad de lectura y escritura) de una tabla de Amazon Keyspaces.	23 de abril de 2020
<a href="#">Nuevo capítulo sobre seguridad</a>	El nuevo capítulo <a href="#">Seguridad</a> de la Guía del usuario de Application Auto Scaling ayuda a entender cómo aplicar el <a href="#">modelo de responsabilidad compartida</a> cuando se usa Application Auto Scaling. En esta actualización, el capítulo “Autenticación y control de acceso” de la guía del usuario se ha sustituido por una nueva sección más sencilla, <a href="#">Identity and Access Management para Application Auto Scaling</a> .	16 de enero de 2020
<a href="#">Actualizaciones menores</a>	Diversas mejoras y correcciones.	15 de enero de 2020

[Adición de la funcionalidad de notificaciones](#)

Application Auto Scaling ahora envía eventos a Amazon EventBridge y notificaciones a usted AWS Health Dashboard cuando se producen determinadas acciones. Para obtener más información, consulte [Monitoreo de Application Auto Scaling](#).

20 de diciembre de 2019

[Añada soporte para AWS Lambda funciones](#)

Utilice el Application Auto Scaling para escalar la simultaneidad aprovisionada de una función de Lambda.

3 de diciembre de 2019

[Agregue compatibilidad con los puntos de conexión de clasificación de documentos de Amazon Comprehend](#)

Utilice Application Auto Scaling para escalar la capacidad de rendimiento de un punto de conexión de clasificación de documentos de Amazon Comprehend.

25 de noviembre de 2019

[Añada compatibilidad con la AppStream versión 2.0 para las políticas de escalado y seguimiento de objetivos](#)

Utilice políticas de escalado y seguimiento de objetivos para ampliar el tamaño de una flota AppStream 2.0.

25 de noviembre de 2019

[Support para puntos de conexión de VPC Amazon](#)

Ahora puede establecer una conexión privada entre Application Auto Scaling VPC y usted. Para obtener información e instrucciones sobre la migración, consulte [Application Auto Scaling y los VPC puntos finales de la interfaz](#).

22 de noviembre de 2019

---

<a href="#">Suspender y reanudar el escalado</a>	Se ha añadido compatibilidad para suspender y reanudar el escalado. Para obtener más información, consulte <a href="#">Suspender y reanudar el escalado para Application Auto Scaling</a> .	29 de agosto de 2019
<a href="#">Sección nueva</a>	Se ha agregado la sección <a href="#">Setting up</a> (Configuración) a la documentación de Application Auto Scaling. Se han realizado mejoras y correcciones menores en toda la guía del usuario.	28 de junio de 2019
<a href="#">Cambios en la guía</a>	Se ha mejorado la documentación de Application Auto Scaling de las secciones <a href="#">Escalado programado</a> , <a href="#">Políticas de escalado por pasos</a> y <a href="#">Políticas de escalado de seguimiento de destino</a> .	11 de marzo de 2019
<a href="#">Agregar compatibilidad con recursos personalizados</a>	Utilice Application Auto Scaling para ampliar recursos personalizados proporcionados por sus propias aplicaciones o servicios. Para obtener más información, consulte nuestro <a href="#">GitHub repositorio</a> .	9 de julio de 2018

[Añada compatibilidad con las variantes de SageMaker terminales](#)

Utilice Application Auto Scaling para escalar el número de instancias de puntos de conexión aprovisionadas para una variante.

28 de febrero de 2018

En la siguiente tabla se describen cambios importantes en la documentación de Application Auto Scaling antes de enero de 2018.

Cambio	Descripción	Fecha
Se ha agregado compatibilidad con réplicas de Aurora.	Utilice el Application Auto Scaling para escalar el número que desee. Para obtener más información, consulte <a href="#">Uso de Amazon Aurora Auto Scaling con réplicas de Aurora</a> en la Guía del RDS usuario de Amazon.	17 de noviembre de 2017
Se ha agregado compatibilidad con acciones de escalado programadas	Utilice el escalado programado para escalar los recursos en horas o intervalos preestablecidos. Para obtener más información, consulte <a href="#">Escalado programado para el Application Auto Scaling</a> .	8 de noviembre de 2017
Se ha añadido compatibilidad con las políticas de escalado de seguimiento de destino	Utilice las políticas de escalado de seguimiento de destino para configurar el escalado dinámico para su aplicación en tan solo unos sencillos pasos. Para obtener más información, consulte <a href="#">Políticas de escalado de</a>	12 de julio de 2017

Cambio	Descripción	Fecha
	<a href="#">seguimiento de destino para Application Auto Scaling</a> .	
Agregue compatibilidad con capacidad de lectura y escritura aprovisionada para tablas e índices secundarios globales de DynamoDB	Utilice Application Auto Scaling para escalar el rendimiento aprovisionado (capacidad de lectura y escritura). Para obtener más información, consulte <a href="#">Administración de la capacidad de rendimiento con el Auto Scaling de DynamoDB</a> en la Guía para desarrolladores de Amazon DynamoDB.	14 de junio de 2017
Añada soporte para flotas AppStream 2.0	Utilice Application Auto Scaling para escalar el tamaño de la flota. Para obtener más información, consulte <a href="#">Fleet Auto Scaling for AppStream 2.0</a> en la Guía de administración de Amazon AppStream 2.0.	23 de marzo de 2017
Añadir soporte para EMR clústeres de Amazon	Utilice el Application Auto Scaling para escalar los nodos principales y de tareas. Para obtener más información, consulta <a href="#">Cómo usar el escalado automático en Amazon EMR</a> en la Guía EMR de administración de Amazon.	18 de noviembre de 2016

Cambio	Descripción	Fecha
Se ha añadido compatibilidad con las flotas de spot	Utilice el Application Auto Scaling para escalar la capacidad de destino. Para obtener más información, consulta <a href="#">Escalado automático o para flotas puntuales</a> en la Guía del EC2 usuario de Amazon.	1 de septiembre de 2016
Añadir soporte para los ECS servicios de Amazon	Utilice el Application Auto Scaling para escalar el número que desee. Para obtener más información, consulte <a href="#">Servicio de escalado automático</a> en la Guía para desarrolladores de servicio de contenedor elástico de Amazon.	9 de agosto de 2016

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