



AWS ParallelCluster Guía del usuario (v3)

AWS ParallelCluster



AWS ParallelCluster: AWS ParallelCluster Guía del usuario (v3)

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 AWS ParallelCluster?	1
Precios	1
Con AWS ParallelCluster figuración	2
Configuración de un Cuenta de AWS	2
Inscríbese en una Cuenta de AWS	2
Creación de un usuario con acceso administrativo	3
Crear un par de claves	4
Instalación de la AWS ParallelCluster CLI	4
Instalar AWS ParallelCluster en un entorno virtual (recomendado)	5
Instalación AWS ParallelCluster en un entorno no virtual mediante pip	7
Instálolo AWS ParallelCluster como una aplicación independiente	8
Pasos que seguir después de la instalación	10
Instalación de la interfaz de usuario de AWS ParallelCluster	10
Instalación de la interfaz de usuario de AWS ParallelCluster	11
Creación de un dominio personalizado	14
Opciones del grupo de usuarios de Amazon Cognito	16
Identificación de la versión de la interfaz de usuario de AWS ParallelCluster y AWS ParallelCluster	19
Actualización de la interfaz de usuario de AWS ParallelCluster a una nueva versión de AWS ParallelCluster	19
Costes de la interfaz de usuario de AWS ParallelCluster	20
Introducción	20
Configurar y crear un clúster con la AWS ParallelCluster CLI	21
Configurar y crear un clúster con la AWS ParallelCluster interfaz de usuario	31
Conectarse a un clúster	33
Acceso de varios usuarios a los clústeres	33
Creación de un Active Directory	34
Creación de un clúster con un dominio de AD	35
Inicio de sesión en un clúster integrado con un dominio de AD	38
Ejecución de trabajos de MPI	39
Ejemplo de configuraciones de clústeres de AWS Managed Microsoft AD sobre LDAP(S)	40
Prácticas recomendadas	44
Prácticas recomendadas: selección del tipo de instancia del nodo principal	44
Prácticas recomendadas: rendimiento de la red	44

Prácticas recomendadas: alertas de presupuesto	46
Mejores prácticas: mover un clúster a una nueva versión AWS ParallelCluster secundaria o a una versión de parche	46
Transición de AWS ParallelCluster 2.x a 3.x	47
Acciones de arranque personalizadas	47
AWS ParallelCluster 2.x y 3.x utilizan una sintaxis de archivo de configuración diferente	48
Lenguaje inclusivo	54
Compatibilidad con el programador	55
AWS ParallelCluster CLI	55
Actualización de la configuración de IMDS	58
Regiones compatibles con AWS ParallelCluster	58
Uso AWS ParallelCluster	60
IU AWS ParallelCluster	60
Configuración de la VPC de AWS Lambda en AWS ParallelCluster	63
AWS Identity and Access Management permisos en AWS ParallelCluster	64
AWS ParallelCluster Funciones de instancia EC2	65
AWS ParallelCluster ejemplos <code>pc1uster</code> de políticas de usuario	66
AWS ParallelCluster ejemplos de políticas de usuario para administrar los recursos de IAM	80
AWS ParallelCluster parámetros de configuración para gestionar los permisos de IAM	87
Configuraciones de red	102
AWS ParallelCluster en una subred pública única	103
AWS ParallelCluster con dos subredes	105
AWS ParallelCluster en una única subred privada conectada mediante AWS Direct Connect	106
AWS ParallelCluster con el programador AWS Batch	107
AWS ParallelCluster en una subred individual sin acceso a Internet	110
Nodos de inicio de sesión	116
Acciones de arranque personalizadas	120
Configuración	122
Arguments	126
Ejemplo de clúster con acciones de arranque personalizadas	126
Ejemplo de actualización de un script de arranque personalizado para IMDSv2	128
Ejemplo de actualización de una configuración para IMDSv1	129
Uso de Amazon S3	130
Ejemplos	130

Uso de instancias de spot	131
Escenario 1: Se interrumpe una instancia de spot sin trabajos en ejecución	131
Escenario 2: Se interrumpe una instancia de spot que ejecuta trabajos de un solo nodo	132
Escenario 3: Se interrumpe una instancia de spot que ejecuta trabajos de varios nodos	132
Programadores compatibles con AWS ParallelCluster	132
Slurm Workload Manager	133
AWS Batch	198
Almacenamiento compartido	206
Configuración del almacenamiento compartido	209
Trabajo con almacenamiento compartido	213
Cuotas	217
Etiquetado	218
Monitoreo de AWS ParallelCluster y registros	221
Integración con Amazon CloudWatch Logs	222
Panel de Amazon CloudWatch	225
Alarmas de Amazon CloudWatch para las métricas del clúster	227
Rotación de registros configurada de AWS ParallelCluster	230
Registros de CLI de <code>pcluster</code>	231
Registros de salida de la consola EC2	232
Recuperar registros AWS ParallelCluster de interfaz de usuario y AWS ParallelCluster tiempo de ejecución	233
Recuperación y conservación de registros	235
AWS CloudFormation recurso personalizado	238
Pila de proveedores alojada por AWS ParallelCluster	239
Recurso del clúster	241
Operaciones de clúster	244
Pilas de solución de problemas que incluyen el recurso AWS ParallelCluster personalizado	244
Elastic Fabric Adapter	245
Habilitar Intel MPI	246
API de AWS ParallelCluster	247
Documentación de la API de AWS ParallelCluster	248
Implementación con AWS CLI	249
Actualización de la API	252
Invocación de la API de AWS ParallelCluster	252
Acceso a registros y métricas de API	255

Conexión al nodo principal a través de NICE DCV	255
Certificado HTTPS NICE DCV	256
Licencias de NICE DCV	256
Uso de <code>pcluster update-cluster</code>	256
Política de actualización: definiciones	257
Ejemplos de <code>pcluster update-cluster</code>	260
AWS ParallelCluster Personalización de AMI	263
AWS ParallelCluster Consideraciones sobre la personalización de la AMI	263
Realice pruebas de validación de componentes personalizadas	264
Supervise el proceso de Generador de imágenes con comandos <code>pcluster</code> para facilitar la depuración	264
Otras consideraciones	265
Inicio de instancias con ODCR (reservas de capacidad bajo demanda)	266
Uso de ODCR con AWS ParallelCluster	266
Parcheo de AMI y sustitución de instancias de EC2	275
Actualización o sustitución de la instancia del nodo principal	277
Guarde los datos de unidades efímeras	277
Detención e inicio del nodo principal de un clúster	277
Sistemas operativos	279
Consideraciones sobre el sistema operativo	280
Referencia para AWS ParallelCluster	282
Comandos de la CLI de AWS ParallelCluster versión 3	282
<code>pcluster</code>	283
<code>pcluster3-config-convert</code>	327
Archivos de configuración	328
Configuración del clúster	328
Cree archivos de configuración de imágenes	463
Referencia de la API de AWS ParallelCluster	471
<code>buildImage</code>	472
<code>createCluster</code>	477
<code>deleteCluster</code>	483
<code>deleteClusterInstances</code>	486
<code>deleteImage</code>	488
<code>describeCluster</code>	491
<code>describeClusterInstances</code>	499
<code>describeComputeFleet</code>	502

describelmage	504
getClusterLogEvents	511
getClusterStackEvents	515
getImageLogEvents	519
getImageStackEvents	523
listClusters	527
listClusterLogStreams	531
listImageLogStreams	535
listImages	539
listOfficialImages	542
updateCluster	545
updateComputeFleet	551
API de la biblioteca Python de AWS ParallelCluster	554
Autorización de las bibliotecas Python de AWS ParallelCluster	554
Instalación de la biblioteca Python de AWS ParallelCluster	554
Operaciones de la API del clúster	555
Computación de las operaciones de la API de flota	558
Operaciones de registro de clústeres y pilas	561
Operaciones de la API de imagen	563
Operaciones de registro de imágenes y pilas	566
Ejemplo	569
AWS Lambda para la biblioteca Python de AWS ParallelCluster	570
Cómo funciona AWS ParallelCluster	572
AWS ParallelCluster procesos	572
clustermgtd	572
clusterstatusmgtd	573
computemgtd	574
Servicios de AWS utilizados por AWS ParallelCluster	574
Amazon API Gateway	575
AWS Batch	575
AWS CloudFormation	575
Amazon CloudWatch	576
CloudWatch Eventos de Amazon	576
Amazon CloudWatch Logs	576
AWS CodeBuild	577
Amazon DynamoDB	577

Amazon Elastic Block Store	577
Amazon Elastic Compute Cloud	577
Amazon Elastic Container Registry	578
Amazon EFS	578
Amazon FSx para Lustre	578
Amazon FSx para ONTAP NetApp	579
Amazon FSx para OpenZFS	579
AWS Identity and Access Management	579
AWS Lambda	580
Amazon RDS	580
Amazon Route 53	580
Amazon Simple Notification Service	580
Amazon Simple Storage Service	580
Amazon VPC	581
Elastic Fabric Adapter	581
EC2 Image Builder	581
NICE DCV	581
AWS ParallelCluster Directorios internos	582
Tutoriales	583
Ejecución del primer trabajo en AWS ParallelCluster	583
Comprobación de la instalación	584
Creación de su primer clúster	584
Iniciar sesión en su nodo principal	585
Ejecución de su primer trabajo con Slurm	586
Creación de una AWS ParallelCluster AMI personalizada	587
Cómo personalizar la AWS ParallelCluster AMI	588
Cree una AWS ParallelCluster AMI personalizada	588
Modificar una AWS ParallelCluster AMI	595
Integración de Active Directory	598
Configuración del cifrado de almacenamiento compartido con una clave AWS KMS	630
Cree la política de	631
Configuración y creación del clúster	632
Ejecución de trabajos en un clúster en modo de cola múltiple	633
Configuración de su clúster	634
Cree su clúster de	635
Inicie sesión en el nodo director	636

Ejecute el trabajo en modo de cola múltiple	637
Uso de la API AWS ParallelCluster	641
Crear un clúster con contabilidad Slurm	655
Paso 1: Crear la VPC y las subredes para AWS ParallelCluster	656
Paso 2: Crear la pila de la base de datos	656
Paso 3: Crear un clúster con la contabilidad Slurm habilitada	657
Volver a una versión anterior del documento de Systems Manager de AWS	657
Volver a una versión anterior del documento SSM	658
Crear un clúster con AWS CloudFormation	660
Creación de clústeres con una pila de creación CloudFormation rápida	661
Creación de clústeres con la interfaz de línea de AWS CloudFormation comandos (CLI)	663
Vea la salida CloudFormation del clúster	665
Acceso al clúster	666
Limpieza	666
AWS ParallelClusterIntegración de la interfaz de usuario con Identity Center	667
Activar IAM Identity Center	667
Añadir su aplicación al centro de identidad de IAM	671
AWS ParallelCluster solución de problemas	679
Intentando crear un clúster	680
failureCode es OnNodeConfiguredExecutionFailure	680
failureCode es OnNodeConfiguredDownloadFailure	680
failureCode es OnNodeConfiguredFailure	681
failureCode es OnNodeStartExecutionFailure	681
failureCode es OnNodeStartDownloadFailure	682
failureCode es OnNodeStartFailure	682
failureCode es EbsMountFailure	683
failureCode es EfsMountFailure	683
failureCode es FsxMountFailure	683
failureCode es RaidMountFailure	684
failureCode es AmiVersionMismatch	684
failureCode es InvalidAmi	684
failureCode está HeadNodeBootstrapFailure con failureReason No se pudo configurar el nodo principal.	685
failureCode está HeadNodeBootstrapFailure agotando el tiempo de espera para la creación del failureReason clúster.	685

failureCode está HeadNodeBootstrapFailure con failureReason No se pudo iniciar el nodo principal.	686
failureCode es ResourceCreationFailure	687
failureCode es ClusterCreationFailure	687
¿Está viendo WaitCondition timed out... en la CloudFormation pila	687
Ver Resource creation cancelled en CloudFormation pila	687
Failed to run cfn-init...¿Ve u otros errores en la AWS CloudFormation pila	688
Visualización de cómo chef-client.log termina con INFO: Waiting for static fleet capacity provisioning	688
Visualización de Failed to run preinstall or postinstall in cfn-init.log	688
¿Está viendo This AMI was created with xxx, but is trying to be used with xxx... en la CloudFormation pila	688
Ver This AMI was not baked by AWS ParallelCluster... en CloudFormation pila	688
Visualización de cómo el comando pcluster create-cluster no se ejecuta localmente	688
Compatibilidad adicional	689
Intentando ejecutar un trabajo	689
srinel trabajo interactivo falla con un error srun: error: fwd_tree_thread: can't find address for <host>, check slurm.conf	689
Job está atascado en el CF estado con squeue el comando	690
Ejecución de trabajos a gran escala y visualización de nfsd: too many open connections, consider increasing the number of threads in /var/log/messages	690
Ejecución de trabajos de MPI	691
Intentando actualizar un clúster	691
pcluster update-clusterel comando no se ejecuta localmente	691
Visualización de clusterStatus como UPDATE_FAILED con el comando pcluster describe-cluster	691
Se agotó el tiempo de espera de la actualización del clúster	692
¿Estás intentando acceder al almacenamiento	692
Uso de un sistema de archivos de Amazon FSx para Lustre externo	692
Uso de un sistema de archivos externo de Amazon Elastic File System	692
¿Está intentando eliminar un clúster	692
El pcluster delete-cluster comando no se puede ejecutar localmente	692

La pila de clústeres no se puede eliminar	693
¿Intentando actualizar la pila AWS ParallelCluster de API	693
Visualización de errores en las inicializaciones de los nodos de computación	693
Visualización de Node bootstrap error en clustermgtd.log	693
Si configuré reservas de capacidad (ODCR) bajo demanda o instancias reservadas zonales	693
Visualización de An error occurred (VcpuLimitExceeded) en slurm_resume.log cuando no puedo ejecutar un trabajo o en clustermgtd.log cuando no puedo crear un clúster	695
Visualización de An error occurred (InsufficientInstanceCapacity) en slurm_resume.log cuando no puedo ejecutar un trabajo o en clustermgtd.log cuando no puedo crear un clúster	695
Visualización de los nodos que están en estado DOWN con Reason (Code:InsufficientInstanceCapacity)... ..	695
Visualización de cannot change locale (en_US.utf-8) because it has an invalid name en slurm_resume.log	696
Ninguno de los escenarios anteriores se aplica a mi situación	696
Solución de problemas de estado del clúster	697
Consulta el gráfico de errores de aprovisionamiento de instancias	697
Ver el gráfico de errores de instancias en mal estado	699
Consulte el gráfico de tiempo de inactividad de Compute Fleet	701
Solución de problemas de implementación del clúster	701
Vea AWS CloudFormation los eventos en CREATE_FAILED	702
Utilice la CLI para ver los flujos de registro	704
Vuelva a crear el clúster fallido con rollback-on-failure	707
Solución de problemas de escalar	708
Registros clave para la depuración	709
Visualización de un error de InsufficientInstanceCapacity en slurm_resume.log cuando no puedo ejecutar un trabajo o en clustermgtd.log cuando no puedo crear un clúster	695
Solución de problemas de inicialización de nodos	711
Solución de problemas de sustituciones y terminaciones inesperadas de nodos	714
Reemplazar, terminar o apagar instancias y nodos problemáticos	715
Estado de la cola (partición) Inactive	716
Solución de otros problemas de nodos y trabajos conocidos	716
Problemas con los grupos de ubicación y el lanzamiento de instancias	716

Directorios que no se pueden reemplazar	716
Solución de problemas de NICE DCV	717
Registros para NICE DCV	717
Problemas con Ubuntu NICE DCV	717
Solución de problemas en clústeres con AWS Batch integración	718
Problemas con el nodo principal	718
Problemas informáticos	718
Errores en los trabajos	718
Error de tiempo de espera de conexión en la URL del punto de conexión	719
Solución de problemas de integración multiusuario con Active Directory	719
Solución de problemas específicos de Active Directory	720
Cómo habilitar el modo de depuración	721
Cómo pasar de LDAPS a LDAP	721
Cómo deshabilitar la verificación del certificado del servidor LDAPS	721
¿Cómo iniciar sesión con una clave SSH en lugar de una contraseña?	722
Cómo restablecer una contraseña de usuario y contraseñas caducadas	722
¿Cómo comprobar el dominio al que se ha unido	723
Cómo solucionar los problemas con	723
Cómo comprobar que la integración con Active Directory funciona	725
Cómo solucionar problemas al iniciar sesión en los nodos de cómputo	726
Problemas conocidos con los trabajos de SimCenter StarCCM+ en un entorno multiusuario	727
Problemas conocidos relacionados con la resolución del nombre de usuario	727
Cómo resolver los problemas de creación del directorio principal	728
Solución de problemas con las AMI de	729
Solución de problemas cuando se agota el tiempo de espera de una actualización del clúster cuando no se está ejecutando cfn-hup	730
Solución de problemas de redes en bastidor	730
Problemas con los clústeres en una única subred pública	730
No se pudo actualizar el clúster al realizar una acción onNodeUpdated personalizada	731
Visualización de errores en la configuración personalizada de Slurm	731
Alarmas de clúster	732
Compatibilidad adicional	732
AWS ParallelCluster política de apoyo	733
Seguridad	734
Información de seguridad de los servicios utilizados por AWS ParallelCluster	735

Protección de datos	735
Cifrado de datos	736
Véase también	738
Identity and Access Management	738
Validación de conformidad	739
Aplicación de TLS 1.2	740
Determinar los protocolos admitidos actualmente	740
Compilar OpenSSL y Python	742
Notas de la versión e historial de revisión	744
.....	dcccxliv

¿Qué es AWS ParallelCluster?

AWS ParallelCluster es una herramienta de administración de clústeres de código abierto compatible con AWS que le ayuda a implementar y administrar clústeres de computación de alto rendimiento (HPC) en la nube de Nube de AWS. Configura automáticamente los recursos informáticos, programador y el sistema de archivos compartido. Puede utilizar AWS ParallelCluster con AWS Batch y Slurm programadores.

Con AWS ParallelCluster, puede crear e implementar rápidamente entornos de cómputo HPC de prueba de concepto y de producción. También puede crear e implementar flujos de trabajo de nivel superior sobre AWS ParallelCluster como, por ejemplo, un portal de genómica que automatice el flujo de trabajo completo de la secuenciación del ADN.

Puede obtener acceso a AWS ParallelCluster utilizando los siguientes métodos:

- [Interfaz de línea de comandos \(CLI\) de AWS ParallelCluster](#)
- [API de AWS ParallelCluster](#)
- [AWS ParallelCluster Interfaz de usuario](#) (agregada con la versión 3.5.0)
- [API de la biblioteca Python de AWS ParallelCluster](#) (agregada con la versión 3.5.0)
- Como [AWS CloudFormation recurso personalizado](#) (agregada con la versión 3.6.0)

Precios

Al utilizar la interfaz de línea de comandos (CLI) o API de AWS ParallelCluster, solo paga por los recursos de AWS que se crean al crear o actualizar imágenes y clústeres de AWS ParallelCluster. Para obtener más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

La interfaz de usuario de AWS ParallelCluster se basa en una arquitectura sin servidor y, en la mayoría de los casos, se puede utilizar en la categoría de nivel AWS gratuito. Para obtener más información, consulte [Costes de la interfaz de usuario de AWS ParallelCluster](#).

Con AWS ParallelCluster figuración

Temas

- [Configuración de un Cuenta de AWS](#)
- [Crear un par de claves](#)
- [Instalación de la interfaz de línea de AWS ParallelCluster comandos \(CLI\)](#)
- [Pasos que seguir después de la instalación](#)
- [Instalación de la interfaz de usuario de AWS ParallelCluster](#)
- [Empezando con AWS ParallelCluster](#)
- [Acceso de varios usuarios a los clústeres](#)
- [Prácticas recomendadas](#)
- [Transición de AWS ParallelCluster 2.x a 3.x](#)
- [Regiones compatibles con AWS ParallelCluster](#)

Configuración de un Cuenta de AWS

Configura una AWS cuenta para usarla AWS ParallelCluster.

Inscríbese en una Cuenta de AWS

Si no tiene una Cuenta de AWS, complete los siguientes pasos para crearlo.

Para suscribirse a una Cuenta de AWS

1. Abra <https://portal.aws.amazon.com/billing/signup>.
2. Siga las instrucciones que se le indiquen.

Parte del procedimiento de registro consiste en recibir una llamada telefónica e indicar un código de verificación en el teclado del teléfono.

Cuando te registras en una Cuenta de AWS, Usuario raíz de la cuenta de AWS se crea una. El usuario raíz tendrá acceso a todos los Servicios de AWS y recursos de esa cuenta. Como práctica recomendada de seguridad, asigne acceso administrativo a un usuario y utilice únicamente el usuario raíz para realizar [tareas que requieren acceso de usuario raíz](#).

AWS te envía un correo electrónico de confirmación una vez finalizado el proceso de registro. Puede ver la actividad de la cuenta y administrar la cuenta en cualquier momento entrando en <https://aws.amazon.com/> y seleccionando Mi cuenta.

Creación de un usuario con acceso administrativo

Después de crear un usuario administrativo Cuenta de AWS, asegúrelo Usuario raíz de la cuenta de AWS AWS IAM Identity Center, habilite y cree un usuario administrativo para no usar el usuario root en las tareas diarias.

Proteja su Usuario raíz de la cuenta de AWS

1. Inicie sesión [AWS Management Console](#) como propietario de la cuenta seleccionando el usuario root e introduciendo su dirección de Cuenta de AWS correo electrónico. En la siguiente página, escriba su contraseña.

Para obtener ayuda para iniciar sesión con el usuario raíz, consulte [Signing in as the root user](#) en la Guía del usuario de AWS Sign-In .

2. Active la autenticación multifactor (MFA) para el usuario raíz.

Para obtener instrucciones, consulte [Habilitar un dispositivo MFA virtual para el usuario Cuenta de AWS raíz \(consola\)](#) en la Guía del usuario de IAM.

Creación de un usuario con acceso administrativo

1. Activar IAM Identity Center.

Consulte las instrucciones en [Activar AWS IAM Identity Center](#) en la Guía del usuario de AWS IAM Identity Center .

2. En IAM Identity Center, conceda acceso administrativo a un usuario.

Para ver un tutorial sobre su uso Directorio de IAM Identity Center como fuente de identidad, consulte [Configurar el acceso de los usuarios con la configuración predeterminada Directorio de IAM Identity Center en la](#) Guía del AWS IAM Identity Center usuario.

Iniciar sesión como usuario con acceso de administrador

- Para iniciar sesión con el usuario de IAM Identity Center, utilice la URL de inicio de sesión que se envió a la dirección de correo electrónico cuando creó el usuario de IAM Identity Center.

Para obtener ayuda para iniciar sesión con un usuario del Centro de identidades de IAM, consulte [Iniciar sesión en el portal de AWS acceso](#) en la Guía del AWS Sign-In usuario.

Concesión de acceso a usuarios adicionales

1. En IAM Identity Center, cree un conjunto de permisos que siga la práctica recomendada de aplicar permisos de privilegios mínimos.

Para conocer las instrucciones, consulte [Create a permission set](#) en la Guía del usuario de AWS IAM Identity Center .

2. Asigne usuarios a un grupo y, a continuación, asigne el acceso de inicio de sesión único al grupo.

Para conocer las instrucciones, consulte [Add groups](#) en la Guía del usuario de AWS IAM Identity Center .

Crear un par de claves

Para implementar clústeres, AWS ParallelCluster lanza instancias EC2 para crear el nodo principal del clúster y los nodos de cómputo. Para realizar tareas del clúster, como ejecutar y monitorear trabajos o administrar usuarios, debe poder acceder al nodo principal del clúster. Para comprobar que puede acceder a la instancia del nodo principal mediante SSH, debe utilizar un par de claves de EC2. Para obtener más información sobre la creación de un par de claves, consulte [Creación de pares de claves](#) en la Guía del usuario de Amazon Elastic Compute Cloud para instancias de Linux.

Instalación de la interfaz de línea de AWS ParallelCluster comandos (CLI)

AWS ParallelCluster se distribuye como un paquete de Python y se instala mediante el administrador de pip paquetes de Python. Para obtener instrucciones sobre cómo instalar paquetes de Python, consulte [Instalación de paquetes](#) en la Guía del usuario de paquetes de Python.

Formas de instalación AWS ParallelCluster:

- [Instalar AWS ParallelCluster en un entorno virtual \(recomendado\)](#)
- [Instalación AWS ParallelCluster en un entorno no virtual mediante pip](#)

- [Instálelo AWS ParallelCluster como una aplicación independiente](#)

Puede encontrar el número de versión de la CLI más reciente en la [página de versiones de GitHub](#). En esta guía, los ejemplos de comandos asumen que ha instalado una versión de Python posterior a la 3.6. Los comandos de ejemplo `pip` utilizan la versión `pip3`.

Administre tanto la AWS ParallelCluster 2 como la AWS ParallelCluster 3

Para los clientes que usan AWS ParallelCluster 2 y AWS ParallelCluster 3 y desean administrar las CLI de ambos paquetes, recomendamos que instalen AWS ParallelCluster 2 y AWS ParallelCluster 3 en diferentes [entornos virtuales](#). Esto garantiza que pueda seguir utilizando todas las versiones de AWS ParallelCluster y todos los recursos del clúster asociados.

Instalar AWS ParallelCluster en un entorno virtual (recomendado)

Se recomienda realizar la instalación AWS ParallelCluster en un entorno virtual para evitar conflictos entre las versiones obligatorias y otros `pip` paquetes.

Requisitos previos

- AWS ParallelCluster requiere Python 3.7 o posterior. Si aún no lo tiene instalado, [descargue una versión compatible](#) para su plataforma en [python.org](#).

Para instalar AWS ParallelCluster en un entorno virtual

1. Si `virtualenv` no está instalado, instale `virtualenv` mediante `pip3`. Si `python3 -m virtualenv help` muestra información de ayuda, vaya al paso 2.

```
$ python3 -m pip install --upgrade pip
$ python3 -m pip install --user --upgrade virtualenv
```

Ejecute `exit` para salir de la ventana de terminal actual y abrir una nueva para detectar los cambios del entorno.

2. Cree un entorno virtual y asígnele un nombre.

```
$ python3 -m virtualenv ~/apc-ve
```

También puede usar la opción `-p` para especificar una versión específica de Python.

```
$ python3 -m virtualenv -p $(which python3) ~/apc-ve
```

3. Active el entorno virtual nuevo.

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

4. Instálelo AWS ParallelCluster en su entorno virtual.

```
(apc-ve)~$ python3 -m pip install --upgrade "aws-parallelcluster"
```

5. Instale Node Version Manager y la última versión de Long-Term Support (LTS) de Node.js. AWS Cloud Development Kit (AWS CDK) (AWS CDK) requiere Node.js CloudFormation para la generación de plantillas.

Note

Si la instalación de Node.js no funciona en su plataforma, puede instalar una versión LTS anterior a la última versión de LTS. Para obtener más información, consulte el [calendario de versiones de Node.js](#) y los requisitos previos del [CDK de AWS](#).

Ejemplo de comando de instalación de Node.js:

```
$ nvm install --lts=Hydrogen
```

```
$ curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash
$ chmod ug+x ~/.nvm/nvm.sh
$ source ~/.nvm/nvm.sh
$ nvm install --lts
$ node --version
```

6. Compruebe que AWS ParallelCluster está instalado correctamente.

```
$ pcluster version
{
  "version": "3.7.0"
}
```

Puede utilizar el comando `deactivate` para salir del entorno virtual. Cada vez que inicie una sesión, debe [activar el entorno de nuevo](#).

Para actualizar a la versión más reciente de AWS ParallelCluster, vuelva a ejecutar el comando de instalación.

```
(apc-ve)~$ python3 -m pip install --upgrade "aws-parallelcluster"
```

Instalación AWS ParallelCluster en un entorno no virtual mediante pip

Requisitos previos

- AWS ParallelCluster requiere Python 3.7 o posterior. Si aún no lo tiene instalado, [descargue una versión compatible](#) para su plataforma en [python.org](#).

Instalar AWS ParallelCluster

1. `pip` Úselo para instalar AWS ParallelCluster.

```
$ python3 -m pip install "aws-parallelcluster" --upgrade --user
```

Cuando utiliza el `--user` conmutador, se `pip` instala en. AWS ParallelCluster `~/.local/bin`

2. Instale Node Version Manager y la última versión de Long-Term Support (LTS) de Node.js. AWS Cloud Development Kit (AWS CDK) (AWS CDK) requiere Node.js CloudFormation para la generación de plantillas.

Note

Si la instalación de Node.js no funciona en su plataforma, puede instalar una versión LTS anterior a la última versión de LTS. Para obtener más información, consulte el [calendario de versiones de Node.js](#) y los requisitos previos del [CDK de AWS](#).

```
$ nvm install --lts=Gallium
```

```
$ curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.38.0/install.sh | bash
$ chmod ug+x ~/.nvm/nvm.sh
```

```
$ source ~/.nvm/nvm.sh
$ nvm install --lts
$ node --version
```

3. Compruebe que se AWS ParallelCluster ha instalado correctamente.

```
$ pcluster version
{
  "version": "3.7.0"
}
```

4. Para actualizar a la versión más reciente, ejecute el comando de instalación de nuevo.

```
$ python3 -m pip install "aws-parallelcluster" --upgrade --user
```

Instálelo AWS ParallelCluster como una aplicación independiente

Instálelo AWS ParallelCluster como una aplicación independiente en su entorno. Siga las instrucciones para realizar la instalación AWS ParallelCluster en un sistema operativo disponible en la siguiente sección.

Requisitos previos

- Un entorno con un sistema operativo compatible con una versión disponible del instalador.

Note

AWS ParallelCluster requiere Nodejs. AWS ParallelCluster El instalador incluye una versión empaquetada de Nodejs (v18), que se instala si aún no existe. Si su sistema no es compatible con NodeJS v18, debe instalar NodeJS antes de instalarlo. AWS ParallelCluster

Linux

Linux x86 (64-bit)

Instálelo en su entorno. AWS ParallelCluster

1. Descargue el [instalador de pcluster](#) más reciente.

2. Descomprima el paquete del instalador e instálelo AWS ParallelCluster mediante los siguientes comandos:

```
$ unzip pcluster-installer-bundle-3.9.1.608-node-v18.17.1-Linux_x86_64-signed.zip
-d pcluster-installer-bundle
$ cd pcluster-installer-bundle
$ chmod +x install_pcluster.sh
```

3. Ejecute el siguiente script de instalación.

```
$ bash install_pcluster.sh
```

4. Compruebe que AWS ParallelCluster está instalado correctamente.

```
$ pcluster version
{
  "version": "3.9.1"
}
```

Solución de errores de la instalación de **pcluster**

- Si la AWS ParallelCluster versión no se devuelve en el paso 4, reinicie el terminal o `source` actualice la `PATH` variable para que incluya el nuevo directorio binario, como se muestra en el siguiente ejemplo: `bash_profile`

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

- Si utiliza la instalación de `pcluster` para crear clústeres con `CustomActions` especificadas como recursos de HTTPS, en lugar de URI de S3, es posible que veas un mensaje de `WARNING` que indique que es posible que estos recursos no estén verificados (`[SSL : CERTIFICATE_VERIFY_FAILED]`). Esto se debe a un problema conocido y puede ignorar esta advertencia si confía en la autenticidad de los recursos especificados.

Versiones anteriores del paquete de instalación

- Ninguna

Pasos que seguir después de la instalación

Puede comprobar que se AWS ParallelCluster ha instalado correctamente ejecutándolo [pcluster version](#).

```
$ pcluster version
{
  "version": "3.7.0"
}
```

AWS ParallelCluster se actualiza periódicamente. Para actualizar a la versión más reciente de AWS ParallelCluster, vuelva a ejecutar el comando de instalación. Para obtener más información sobre la última versión de AWS ParallelCluster, consulte las [notas AWS ParallelCluster de la versión](#).

```
$ pip3 install aws-parallelcluster --upgrade --user
```

Para AWS ParallelCluster desinstalarla, utilice `pip3 uninstall`.

```
$ pip3 uninstall aws-parallelcluster
```

Si no dispone de Python ni `pip3`, utilice el procedimiento para su entorno.

Instalación de la interfaz de usuario de AWS ParallelCluster

La interfaz de usuario de AWS ParallelCluster es una interfaz de usuario basada en la web que refleja la CLI `pcluster` de AWS ParallelCluster y, al mismo tiempo, proporciona una experiencia similar a la de una consola. La interfaz de usuario de AWS ParallelCluster se instala en la Cuenta de AWS, desde donde también puede acceder a ella. Cuando la ejecuta, la interfaz de usuario de AWS ParallelCluster accede a una instancia de la API de AWS ParallelCluster alojada en Amazon API Gateway en la Cuenta de AWS. Para obtener más información acerca de la interfaz de usuario de AWS ParallelCluster, consulte [IU AWS ParallelCluster](#).

Requisitos previos:

- Una Cuenta de AWS
- Acceso a la AWS Management Console

Temas

- [Instalación de la interfaz de usuario de AWS ParallelCluster](#)
- [Creación de un dominio personalizado](#)
- [Opciones del grupo de usuarios de Amazon Cognito](#)
- [Identificación de la versión de la interfaz de usuario de AWS ParallelCluster y AWS ParallelCluster](#)
- [Actualización de la interfaz de usuario de AWS ParallelCluster a una nueva versión de AWS ParallelCluster](#)
- [Costes de la interfaz de usuario de AWS ParallelCluster](#)

Instalación de la interfaz de usuario de AWS ParallelCluster

Para instalar una instancia de la interfaz de usuario de AWS ParallelCluster, elija un enlace de creación rápida de AWS CloudFormation para la Región de AWS en la que cree los clústeres. La URL de creación rápida le lleva a un asistente de creación de pilas, en el que puede proporcionar entradas de las plantillas de creación rápida de pilas e implementar la pila. Para obtener más información sobre la creación CloudFormation rápida de pilas, consulte [Creación de enlaces de creación rápida para pilas](#) en la Guía del usuario. AWS CloudFormation

Note

Solo puede crear y editar clústeres o crear imágenes con la misma versión de AWS ParallelCluster que ha utilizado para instalar la interfaz de usuario de AWS ParallelCluster.

Enlaces de creación rápida de la interfaz de usuario de AWS ParallelCluster por región

Enlaces creación rápida de la interfaz de usuario

[us-east-1](#)

[us-east-2](#)

[us-west-1](#)

[us-west-2](#)

[eu-west-1](#)

Enlaces creación rápida de la interfaz de usuario

[eu-west-2](#)

[eu-west-3](#)

[eu-central-1](#)

[eu-north-1](#)

[me-south-1](#)

[sa-east-1](#)

[ca-central-1](#)

[ap-northeast-1](#)

[ap-northeast-2](#)

[ap-south-1](#)

[ap-southeast-1](#)

[ap-southeast-2](#)

[us-gov-west-1](#)

Utilice un enlace de creación rápida de AWS CloudFormation para implementar una pila de la interfaz de usuario de AWS ParallelCluster con pilas anidadas de Amazon Cognito, API Gateway y Amazon EC2 Systems Manager.

1. Inicie sesión en la AWS Management Console.
2. Para implementar la interfaz de usuario de AWS ParallelCluster, elija un enlace de creación rápida de Región de AWS en la tabla que aparece al principio de esta sección. Esto le llevará al asistente de CloudFormation creación de pilas de la consola.
3. Introduzca una dirección de correo electrónico válida para el correo electrónico del administrador.

Una vez que la implementación se complete correctamente, la interfaz de usuario de AWS ParallelCluster le enviará una contraseña temporal a esta dirección de correo electrónico. La contraseña temporal se utiliza para acceder a la interfaz de usuario de AWS ParallelCluster. Si elimina el correo electrónico antes de guardar o usar la contraseña temporal, debe eliminar la pila y volver a instalar la interfaz de usuario de AWS ParallelCluster.

4. Deje el resto del formulario en blanco o introduzca valores en los parámetros (opcionales) para personalizar la creación de la interfaz de usuario de AWS ParallelCluster.
5. Tenga en cuenta el nombre de la pila para usarlo en pasos posteriores.
6. Navegue hasta Capacidades. Acepte las CloudFormation capacidades.
7. Seleccione Crear. La implementación de la interfaz de usuario de AWS ParallelCluster y la API de AWS ParallelCluster tarda unos 15 minutos en completarse.
8. Vea los detalles de la pila a medida que se crea la pila.
9. Una vez completada la implementación, abra el correo electrónico de administrador enviado a la dirección que ha introducido. Contiene una contraseña temporal que se utiliza para obtener acceso a la interfaz de usuario de AWS ParallelCluster. Si elimina el correo electrónico de forma permanente y aún no ha iniciado sesión en la interfaz de usuario de AWS ParallelCluster, debe eliminar la pila de la interfaz de usuario de AWS ParallelCluster que haya creado y volver a instalar la interfaz de usuario de AWS ParallelCluster.
10. En la lista de pilas de la consola AWS CloudFormation, elija el enlace al nombre de pila que anotó en un paso anterior.
11. En Detalles de la pila, elija Salidas y seleccione el enlace de la clave denominada **Stackname**URL para abrir la interfaz de usuario de AWS ParallelCluster. **Stackname** es el nombre que anotó en un paso anterior.
12. Introduzca la contraseña temporal. Siga los pasos para crear su propia contraseña e iniciar sesión.
13. Ahora se encuentra en la página de inicio de la interfaz de usuario de AWS ParallelCluster en la Región de AWS que ha seleccionado.
14. Para comenzar a utilizar la interfaz de usuario de AWS ParallelCluster, consulte [Configuración y creación de un clúster con la interfaz de usuario de AWS ParallelCluster](#).

Note

Las sesiones de PCUI tienen una duración predeterminada de 5 minutos, que es el valor mínimo proporcionado por Cognito a partir de la PCUI 2023.12.0. Por lo tanto, se espera que un usuario eliminado de los grupos de usuarios de Cognito pueda seguir accediendo al sistema hasta que caduque la sesión.

Creación de un dominio personalizado

Obtenga información sobre cómo crear un dominio personalizado para la interfaz de usuario de AWS ParallelCluster. La interfaz de usuario está alojada en Amazon API Gateway en su Cuenta de AWS. Puede crear un nombre de dominio personalizado en la consola de API Gateway.

Requisitos previos:

- Tener una Cuenta de AWS.
- Tener una instancia de la interfaz de usuario de AWS ParallelCluster a la que se pueda acceder.
- Ser propietario de un dominio.
- Poder cambiar la configuración básica del sistema de nombres de dominio (DNS).

Paso 1: crear un dominio nuevo en Amazon API Gateway

1. En AWS Management Console, vaya a [API Gateway](#), donde podrá ver la API de su interfaz de usuario de AWS ParallelCluster.
2. En el panel de navegación, elija Custom Domain Names (Nombres de dominio personalizados).
3. Seleccione Crear.
4. En Detalles del dominio, introduzca el nombre de su dominio.
5. En Configuración de punto de conexión, elija un certificado ACM existente o elija Crear un nuevo certificado ACM.

(Opcional) Cree un certificado

- a. En la consola ACM, elija Solicitar.
- b. En Nombres de dominio, escriba el nombre de su dominio.
- c. En Método de validación, elija un método de validación.

Si elige Validación por email, se envía un correo electrónico a la dirección de correo electrónico que figura en el registrador de dominios.

- d. Seleccione Apruebo para activar el certificado.

Paso 2: configurar mapeos de API

1. En [API Gateway](#), Nombres de dominio personalizados your-domain-name, elija Configurar asignaciones de API.
2. Elija Custom domain names (Nombres de dominio personalizados).
3. Seleccione Add new mapping (Agregar nueva asignación).
4. Elija la API de la interfaz de usuario de AWS ParallelCluster, la Etapa \$default y Guardar.
5. En Nombre de dominio de API Gateway, copie el valor para usarlo en los pasos siguientes.

Paso 3: configurar DNS

- Cree una regla CNAME de DNS que dirija su dominio al dominio de API Gateway. Introduzca solo el dominio. Por ejemplo, no añada la etapa, como beta o prod. Sustituya *abcde12345* por su ID de API Gateway y sustituya *us-east-2* por la Región de AWS de la API.

Regla	Origen	Destino
CNAME	<i>example.com</i>	d- <i>abcde12345</i> .execute-api. <i>us-east-2</i> .amazonaws.com

Paso 4: agregar un dominio a su grupo de usuarios de Amazon Cognito

1. Vaya a la [consola de Amazon Cognito](#).
2. Seleccione el enlace del grupo de usuarios.
3. Seleccione Integración de aplicaciones.
4. In Dominio, elija Acciones, Crear dominio personalizado.
5. Introduzca su Dominio personalizado y seleccione su Certificado ACM.
6. Elija Crear nombre de dominio personalizado.

Paso 5: configurar la URL de devolución de llamada de API Gateway

1. Vaya a la [consola de Amazon Cognito](#).
2. En Integración de aplicaciones, Clientes de aplicaciones y análisis del grupo de usuarios de Amazon Cognito, seleccione el enlace de la aplicación.
3. En Interfaz de usuario alojada, seleccione Editar.
4. En URL de devolución de llamada permitidas, seleccione Agregar otra URL e introduzca una URL de devolución de llamada, por ejemplo `example.com/login`.

Paso 6: configurar la función de Lambda

1. Vaya a la [consola Lambda](#).
2. Seleccione Funciones en el panel de navegación.
3. Filtre la lista de funciones para buscar la `ParallelClusterUIFunction` y seleccione el enlace.
4. Elija Configuración, Variables de entorno.
5. Elija Editar.
6. Para el valor `SITE_URL`, introduzca su dominio personalizado.
7. Navegue hasta su dominio, por ejemplo `example.com`, y auténtíquese para conectarse a la interfaz de usuario de AWS ParallelCluster.

Opciones del grupo de usuarios de Amazon Cognito

Las siguientes secciones hacen referencia a enlaces de creación CloudFormation rápida o URL de creación rápida. La URL de creación rápida le lleva a un asistente de creación de pilas, en el que puede proporcionar entradas de las plantillas de creación rápida de pilas e implementar la pila. Para obtener más información sobre la creación CloudFormation rápida de pilas, consulte Creación de enlaces de [creación rápida](#) para pilas en la Guía del usuario. AWS CloudFormation

Para mantener un grupo de usuarios de Amazon Cognito que pueda usar con varias instancias de la interfaz de usuario de AWS ParallelCluster, tenga en cuenta las siguientes opciones:

- Utilice una instancia de AWS ParallelCluster interfaz de usuario existente que se vincule a un grupo de usuarios de Amazon Cognito creado a partir de una pila CloudFormation anidada. Esto

es lo que se crea al implementar la interfaz de usuario de AWS ParallelCluster mediante el enlace de creación rápida y mantener en blanco todos los parámetros de Amazon Cognito.

- Utilice un grupo de usuarios de Amazon Cognito independiente que se implemente antes de implementar la interfaz de usuario de AWS ParallelCluster. A continuación, implemente una nueva instancia de la interfaz de usuario de AWS ParallelCluster vinculada al grupo de usuarios independiente de Amazon Cognito que ya ha implementado. De esta forma, se separa la implementación de Amazon Cognito de la implementación de la interfaz de usuario de AWS ParallelCluster. Además, las CloudFormation pilas de AWS ParallelCluster interfaz de usuario no anidadas son más fáciles de actualizar.

Uso de un grupo de usuarios de Amazon Cognito existente con una nueva instancia de la interfaz de usuario de AWS ParallelCluster

1. En la CloudFormation consola, seleccione la pila de AWS ParallelCluster interfaz de usuario que contiene el grupo de usuarios de Amazon Cognito que desee utilizar con varias instancias de AWS ParallelCluster interfaz de usuario.
2. Navegue hasta la pila anidada que creó el grupo de usuarios de Amazon Cognito.
3. Seleccione la pestaña Salidas.
4. Copie los valores de los siguientes parámetros:
 - `UserPoolId`
 - `UserPoolAuthDomain`
 - `SNSRole`
5. Implemente una nueva instancia de la interfaz de usuario de AWS ParallelCluster mediante el enlace de creación rápida y rellene todos los parámetros de `External AWS ParallelCluster UI Amazon Cognito` con los resultados que ha copiado. Esto evita que la nueva pila de la interfaz de usuario de AWS ParallelCluster cree un nuevo grupo y lo vincula al grupo de usuarios de Amazon Cognito existente que se creó a partir de una pila anidada. Puede implementar nuevas instancias de la interfaz de usuario de AWS ParallelCluster subsiguientes que tengan los mismos valores de parámetros y puede vincularlas al grupo de usuarios de Amazon Cognito.

Creación de un grupo de usuarios de Amazon Cognito independiente

Enlaces de creación rápida de Amazon Cognito de la interfaz de usuario de AWS ParallelCluster por región

Enlaces de creación rápida de Amazon Cognito de la interfaz de usuario

[us-east-1](#)

[us-east-2](#)

[us-west-1](#)

[us-west-2](#)

[eu-west-1](#)

[eu-west-2](#)

[eu-west-3](#)

[eu-central-1](#)

[eu-north-1](#)

[me-south-1](#)

[sa-east-1](#)

[ca-central-1](#)

[ap-northeast-1](#)

[ap-northeast-2](#)

[ap-south-1](#)

[ap-southeast-1](#)

[ap-southeast-2](#)

[us-gov-west-1](#)

1. Para lanzar una pila exclusiva de Amazon Cognito, elija un enlace de creación rápida etiquetado con la misma Región de AWS en la que implemente sus instancias de la interfaz de usuario de AWS ParallelCluster. Consulte los enlaces de creación rápida al principio de esta sección.
2. Una vez finalizada la creación de la pila, seleccione la pestaña Salidas y copie los valores de los siguientes parámetros:
 - `UserPoolId`
 - `UserPoolAuthDomain`
 - `SNSRole`
3. Implemente una nueva instancia de la interfaz de usuario de AWS ParallelCluster seleccionando un enlace de inicio rápido de la interfaz de usuario de AWS ParallelCluster y rellenando todos los parámetros de `External AWS ParallelCluster UI Amazon Cognito` con los valores que ha copiado. La nueva instancia de la interfaz de usuario de AWS ParallelCluster se vincula con el grupo de usuarios independiente de Amazon Cognito y no crea una pila anidada ni un grupo de usuarios nuevo. Puede implementar nuevas instancias de la interfaz de usuario de AWS ParallelCluster subsiguientes que tengan los mismos valores de parámetros y puede vincularlas al grupo de usuarios de Amazon Cognito independiente.

Identificación de la versión de la interfaz de usuario de AWS ParallelCluster y AWS ParallelCluster

1. En la CloudFormation consola, selecciona una pila de AWS ParallelCluster interfaz de usuario.
2. Seleccione la pestaña Parámetros.
3. La versión de AWS ParallelCluster es el valor del parámetro Versión.
4. La versión de la AWS ParallelCluster interfaz de usuario se encuentra al final del `PublicEcrImageUri` valor. Por ejemplo, si el valor es `public.ecr.aws/pcui/parallelcluster-ui-awslambda:2023.02`, entonces la versión es `2023.02`.

Actualización de la interfaz de usuario de AWS ParallelCluster a una nueva versión de AWS ParallelCluster

Para actualizar la interfaz de usuario de AWS ParallelCluster a la versión de AWS ParallelCluster más reciente, seleccione un [enlace de creación rápida](#) para lanzar una nueva pila.

Costes de la interfaz de usuario de AWS ParallelCluster

La interfaz de usuario de AWS ParallelCluster se basa en una arquitectura sin servidor y, en la mayoría de los casos, se puede utilizar en la categoría de nivel AWS gratuito. En la siguiente tabla se enumeran los Servicios de AWS de los que depende la interfaz de usuario de AWS ParallelCluster y sus límites de nivel gratuito. Se estima que el uso típico cuesta menos de un dólar al mes.

Servicio	Nivel gratuito de AWS
Amazon Cognito	50 000 usuarios activos mensuales
Amazon API Gateway	1 millón de llamadas a la API REST
AWS Lambda	1 millón de solicitudes gratuitas y 400 000 GB por segundo de tiempo de procesamiento cada mes
EC2 Image Builder	Sin coste alguno, excepto EC2
Amazon Elastic Compute Cloud	Creación de una imagen de contenedor única de 15 minutos
AWS CloudFormation	5 GB de datos (ingesta, almacenamiento de archivos y datos escaneados mediante consultas de Logs Insights)

Empezando con AWS ParallelCluster

Comience por configurar y crear un clúster mediante la interfaz de línea de AWS ParallelCluster comandos (CLI) o la interfaz de usuario (UI) basada en web. La AWS ParallelCluster interfaz de usuario se agregó en la versión 3.5.0.

Temas

- [Configure y cree un clúster con la interfaz de línea de AWS ParallelCluster comandos](#)
- [Configuración y creación de un clúster con la interfaz de usuario de AWS ParallelCluster](#)
- [Conectarse a un clúster](#)

Configure y cree un clúster con la interfaz de línea de AWS ParallelCluster comandos

Tras la instalación AWS ParallelCluster, complete los siguientes pasos de configuración.

Compruebe que su AWS cuenta tenga una función que incluya los permisos necesarios para ejecutar la `pcluster` CLI. Para obtener más información, consulte [AWS ParallelCluster ejemplos pcluster de políticas de usuario](#).

Configure sus AWS credenciales. Para obtener más información, consulte [Configuración de AWS CLI](#) en la Guía del usuario de AWS CLI .

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
Default region name [us-east-1]: us-east-1
Default output format [None]:
```

El Región de AWS lugar donde se lanza el clúster debe tener al menos un par de claves de Amazon EC2. Para obtener más información, consulte [Pares de claves de Amazon EC2](#) en la Guía del usuario de Amazon EC2 para instancias Linux.

Al utilizar la interfaz de línea de AWS ParallelCluster comandos (CLI), solo paga por los AWS recursos que se crean al crear o actualizar AWS ParallelCluster imágenes y clústeres. Para obtener más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

Creación y configuración de su primer clúster

Cree su primer clúster mediante el comando CLI de `pcluster configure` para iniciar un asistente que le solicitará toda la información necesaria para configurar y crear el clúster. Los detalles de la secuencia difieren cuando se usa AWS Batch como programador en comparación con cuando se usa Slurm.

Slurm

```
$ pcluster configure --config config-file.yaml
```

En la lista de Región de AWS identificadores válidos, elija el Región de AWS lugar en el que desea que se ejecute el clúster.

Note

La lista que Regiones de AWS se muestra se basa en la partición de su cuenta y solo incluye las Regiones de AWS que están habilitadas para su cuenta. Para obtener más información sobre cómo habilitar Regiones de AWS su cuenta, consulte [Administrar Regiones de AWS](#) en Referencia general de AWS. El ejemplo que se muestra es de la partición AWS global. Si su cuenta está en la AWS GovCloud (US) partición, solo aparecerá Regiones de AWS en esa partición (gov-us-east-1gov-us-west-1). Del mismo modo, si tu cuenta está en la partición de AWS China, solo cn-north-1 cn-northwest-1 se muestran. Para ver la lista completa de las Regiones de AWS admitidas por AWS ParallelCluster, consulte [Regiones compatibles con AWS ParallelCluster](#).

Allowed values for Región de AWS ID:

1. af-south-1
2. ap-east-1
3. ap-northeast-1
4. ap-northeast-2
5. ap-south-1
6. ap-southeast-1
7. ap-southeast-2
8. ca-central-1
9. eu-central-1
10. eu-north-1
11. eu-south-1
12. eu-west-1
13. eu-west-2
14. eu-west-3
15. me-south-1
16. sa-east-1
17. us-east-1
18. us-east-2
19. us-west-1
20. us-west-2

Región de AWS ID [ap-northeast-1]:

El par de claves se selecciona entre los pares de claves que están registrados con Amazon EC2 en la Región de AWS seleccionada. Elija el par de claves:

```
Allowed values for EC2 Key Pair Name:
```

1. your-key-1
2. your-key-2

```
EC2 Key Pair Name [your-key-1]:
```

Elija el programador que desea utilizar con el clúster.

```
Allowed values for Scheduler:
```

1. slurm
2. awsbatch

```
Scheduler [slurm]:
```

Elija el sistema operativo.

```
Allowed values for Operating System:
```

1. alinux2
2. centos7
3. ubuntu2204
4. ubuntu2004
5. rhel8

```
Operating System [alinux2]:
```

Elija el tipo de instancia del nodo principal:

```
Head node instance type [t2.micro]:
```

Elija la configuración de la cola. Nota: El tipo de instancia no se puede especificar para varios recursos de computación de la misma cola.

```
Number of queues [1]:
```

```
Name of queue 1 [queue1]:
```

```
Number of compute resources for queue1 [1]: 2
```

```
Compute instance type for compute resource 1 in queue1 [t2.micro]:
```

```
Maximum instance count [10]:
```

Permita que EFA ejecute aplicaciones que requieran altos niveles de comunicación entre instancias de forma escalable y sin AWS coste adicional:

- Elija un tipo de instancia que [sea compatible con Elastic Fabric Adapter \(EFA\)](#)..

- Habilite [EFA](#).
- Especifique un nombre de [Grupo de ubicación](#) existente. Si lo deja en blanco, AWS ParallelCluster crea uno automáticamente.

```
Compute instance type for compute resource 2 in queue1 [t2.micro]: c5n.18xlarge
Enable EFA on c5n.18xlarge (y/n) [y]: y
Maximum instance count [10]:
Placement Group name []:
```

Una vez completados los pasos anteriores, decida si desea utilizar una VPC existente o dejar que AWS ParallelCluster cree una VPC por usted. Si no tiene una VPC configurada correctamente, AWS ParallelCluster puede crear una nueva para usted. Coloca tanto el nodo principal como los de computación de la misma subred pública, o solo el nodo principal en una subred pública con todos los nodos de computación en una subred privada. Si permite AWS ParallelCluster crear una VPC, debe decidir si todos los nodos deben estar en una subred pública. Para obtener más información, consulte [Configuraciones de red](#).

Si configura el clúster para usar tipos de instancias que tienen varias interfaces de red o una tarjeta de red, consulte [Configuraciones de red](#) para conocer los requisitos de red adicionales.

Es posible alcanzar la cuota del número de VPC permitidas en una Región de AWS. La cuota predeterminada es de cinco VPC para una Región de AWS. Para obtener más información acerca de esta cuota y cómo solicitar un aumento, consulte [VPC y subredes](#) en la Guía del usuario de Amazon VPC.

Important

Las VPC creadas por AWS ParallelCluster no habilitan los registros de flujo de VPC de forma predeterminada. Los registros de flujo de VPC le permiten capturar información sobre el tráfico IP entrante y saliente de las interfaces de red de su VPC. Para obtener más información, consulte [Logs de flujo de VPC](#) en la Guía del usuario de Amazon VPC.

Si permite AWS ParallelCluster crear una VPC, asegúrese de decidir si todos los nodos deben estar en una subred pública.

Note

Si escoge 1. Head node in a public subnet and compute fleet in a private subnet, AWS ParallelCluster crea una puerta de enlace NAT que genera un coste adicional, incluso si especifica los recursos de nivel gratuito.

```
Automate VPC creation? (y/n) [n]: y
Allowed values for Availability Zone:
1. us-east-1a
2. us-east-1b
3. us-east-1c
4. us-east-1d
5. us-east-1e
6. us-east-1f
Availability Zone [us-east-1a]:
Allowed values for Network Configuration:
1. Head node in a public subnet and compute fleet in a private subnet
2. Head node and compute fleet in the same public subnet
Network Configuration [Head node in a public subnet and compute fleet in a private
subnet]: 1
Beginning VPC creation. Please do not leave the terminal until the creation is
finalized
```

Si no crea una VPC nueva, debe seleccionar una VPC existente.

Si opta por AWS ParallelCluster crear la VPC, anote el ID de la VPC para poder usarlo para AWS CLI eliminarlo más adelante.

```
Automate VPC creation? (y/n) [n]: n
Allowed values for VPC ID:
# id name number_of_subnets
---
1 vpc-0b4ad9c4678d3c7ad ParallelClusterVPC-20200118031893 2
2 vpc-0e87c753286f37eef ParallelClusterVPC-20191118233938 5
VPC ID [vpc-0b4ad9c4678d3c7ad]: 1
```

Una vez seleccionada la VPC, decida si desea utilizar subredes existentes o crear nuevas.

```
Automate Subnet creation? (y/n) [y]: y
```

```
Creating CloudFormation stack...  
Do not leave the terminal until the process has finished
```

AWS Batch

```
$ pcluster configure --config config-file.yaml
```

En la lista de Región de AWS identificadores válidos, elige el Región de AWS lugar en el que quieres que se ejecute el clúster.

Note

La lista que Regiones de AWS se muestra se basa en la partición de su cuenta. Solo incluye las Regiones de AWS que están habilitadas para su cuenta. Para obtener más información sobre cómo habilitar Regiones de AWS su cuenta, consulte [Administrar Regiones de AWS](#) en Referencia general de AWS. El ejemplo que se muestra es de la partición AWS global. Si su cuenta está en la AWS GovCloud (US) partición, solo aparecerá Regiones de AWS en esa partición (gov-us-east-1gov-us-west-1). Del mismo modo, si tu cuenta está en la partición de AWS China, solo cn-north-1 cn-northwest-1 se muestran. Para ver la lista completa de las Regiones de AWS admitidas por AWS ParallelCluster, consulte [Regiones compatibles con AWS ParallelCluster](#).

Allowed values for Región de AWS ID:

1. af-south-1
2. ap-east-1
3. ap-northeast-1
4. ap-northeast-2
5. ap-south-1
6. ap-southeast-1
7. ap-southeast-2
8. ca-central-1
9. eu-central-1
10. eu-north-1
11. eu-south-1
12. eu-west-1
13. eu-west-2
14. eu-west-3

```
15. me-south-1
16. sa-east-1
17. us-east-1
18. us-east-2
19. us-west-1
20. us-west-2
Región de AWS ID [us-east-1]:
```

El par de claves se selecciona entre los pares de claves registrados con Amazon EC2 en la Región de AWS seleccionada. Elija el par de claves:

```
Allowed values for EC2 Key Pair Name:
1. your-key-1
2. your-key-2
EC2 Key Pair Name [your-key-1]:
```

Elija el programador que desea utilizar con el clúster.

```
Allowed values for Scheduler:
1. slurm
2. awsbatch
Scheduler [slurm]: 2
```

Cuando `awsbatch` se selecciona como programador, `alinux2` se utiliza como sistema operativo. Se especifica el tipo de instancia del nodo principal:


```
Head node instance type [t2.micro]:
```

Elija la configuración de la cola. El AWS Batch programador solo contiene una cola única. Se especifica el tamaño máximo del clúster de nodos de computación. Esto se mide en número de vCPU.

```
Number of queues [1]:
Name of queue 1 [queue1]:
Maximum vCPU [10]:
```


Decida si desea utilizar las VPC existentes o dejar que las AWS ParallelCluster cree por usted. Si no tiene una VPC configurada correctamente, AWS ParallelCluster puede crear una nueva. Usa tanto el nodo principal como los de computación de la misma subred pública, o solo el nodo

principal en una subred pública con todos los nodos en una subred privada. Es posible alcanzar la cuota del número de VPC permitidas en una región. El número predeterminado de VPC es cinco. Para obtener más información acerca de esta cuota y cómo solicitar un aumento, consulte [VPC y subredes](#) en la Guía del usuario de Amazon VPC.

 Important

Las VPC creadas por AWS ParallelCluster no habilitan los registros de flujo de VPC de forma predeterminada. Los registros de flujo de VPC le permiten capturar información sobre el tráfico IP entrante y saliente de las interfaces de red de su VPC. Para obtener más información, consulte [Logs de flujo de VPC](#) en la Guía del usuario de Amazon VPC.

Si permite AWS ParallelCluster crear una VPC, asegúrese de decidir si todos los nodos deben estar en una subred pública.

 Note

Si escoge 1. Head node in a public subnet and compute fleet in a private subnet, AWS ParallelCluster crea una puerta de enlace NAT que genera un coste adicional, incluso si especifica los recursos de nivel gratuito.

```
Automate VPC creation? (y/n) [n]: y
Allowed values for Availability Zone:
1. us-east-1a
2. us-east-1b
3. us-east-1c
4. us-east-1d
5. us-east-1e
6. us-east-1f
Availability Zone [us-east-1a]:
Allowed values for Network Configuration:
1. Head node in a public subnet and compute fleet in a private subnet
2. Head node and compute fleet in the same public subnet
Network Configuration [Head node in a public subnet and compute fleet in a private
subnet]: *1*
Beginning VPC creation. Please do not leave the terminal until the creation is
finalized
```

Si no crea una VPC nueva, debe seleccionar una VPC existente.

Si decide AWS ParallelCluster crear la VPC, anote el ID de la VPC para poder usarlo AWS CLI o AWS Management Console eliminarlo más adelante.

```
Automate VPC creation? (y/n) [n]: n
Allowed values for VPC ID:
#  id                                     name                                     number_of_subnets
---  -----
 1  vpc-0b4ad9c4678d3c7ad  ParallelClusterVPC-20200118031893      2
 2  vpc-0e87c753286f37eef  ParallelClusterVPC-20191118233938      5
VPC ID [vpc-0b4ad9c4678d3c7ad]: 1
```

Una vez seleccionada la VPC, decida si desea utilizar subredes existentes o crear nuevas.

```
Automate Subnet creation? (y/n) [y]: y
```

```
Creating CloudFormation stack...
Do not leave the terminal until the process has finished
```

Cuando haya completado los pasos anteriores, se inicia un clúster simple en una VPC. La VPC usa una subred existente que admite direcciones IP públicas. La tabla de enrutamiento de la subred es `0.0.0.0/0 => igw-xxxxxx`. Tenga en cuenta las siguientes condiciones:

- La VPC debe tener `DNS Resolution = yes` y `DNS Hostnames = yes`.
- La VPC también deben tener las opciones de DHCP con el `domain-name` correcto para la Región de AWS. El conjunto de opciones de DHCP predeterminado ya especifica el `AmazonProvidedDNS` necesario. Si especifica más de un servidor de nombres de dominio, consulte [Conjuntos de opciones de DHCP](#) en la Guía del usuario de Amazon VPC. Cuando utilice subredes privadas, utilice una puerta de enlace NAT o un proxy interno para habilitar el acceso web a los nodos de computación. Para obtener más información, consulte [Configuraciones de red](#).

Cuando todas las opciones tengan valores válidos, podrá lanzar el clúster ejecutando el comando `create`.

```
$ pcluster create-cluster --cluster-name test-cluster --cluster-configuration cluster-config.yaml
```

```
{
  "cluster": {
    "clusterName": "test-cluster",
    "cloudformationStackStatus": "CREATE_IN_PROGRESS",
    "cloudformationStackArn": "arn:aws:cloudformation:eu-west-1:xxx:stack/test-cluster/
abcdef0-f678-890a-5abc-021345abcdef",
    "region": "eu-west-1",
    "version": "3.7.0",
    "clusterStatus": "CREATE_IN_PROGRESS"
  },
  "validationMessages": []
}
```

Siga el progreso del clúster:

```
$ pcluster describe-cluster --cluster-name test-cluster
```

o

```
$ pcluster list-clusters --query 'clusters[?clusterName==`test-cluster`]'
```

Una vez que el clúster llegue al estado "clusterStatus": "CREATE_COMPLETE", podrá conectarse a él utilizando su configuración del cliente de SSH normal. Para obtener más información sobre la conexión a instancias de Amazon EC2, consulte la Guía del usuario de [EC2 en la Guía del usuario](#) de Amazon EC2. O bien, puede conectar el clúster mediante

```
$ pcluster ssh --cluster-name test-cluster -i ~/path/to/keyfile.pem
```

Ejecute el siguiente comando para eliminar el clúster.

```
$ pcluster delete-cluster --region us-east-1 --cluster-name test-cluster
```

Una vez eliminado el clúster, puede eliminar los recursos de red de la VPC eliminando la pila de CloudFormation redes. El nombre de la pila comienza por "parallelclusternetworking-" y contiene la hora de creación en formato "AAAAMMDDHHMMSS". Puede enumerar las pilas mediante el comando [list-stacks](#).

```
$ aws --region us-east-1 cloudformation list-stacks \
  --stack-status-filter "CREATE_COMPLETE" \
```

```
--query "StackSummaries[].StackName" | \  
grep -e "parallelclusternetworking-" \  
"parallelclusternetworking-pubpriv-20191029205804"
```

La pila se puede eliminar mediante el comando [delete-stack](#).

```
$ aws --region us-east-1 cloudformation delete-stack \  
--stack-name parallelclusternetworking-pubpriv-20191029205804
```

La VPC que [pcluster configure](#) crea para usted no se crea en la pila de CloudFormation redes. Puede eliminar esa VPC manualmente en la consola o mediante AWS CLI.

```
$ aws --region us-east-1 ec2 delete-vpc --vpc-id vpc-0b4ad9c4678d3c7ad
```

Configuración y creación de un clúster con la interfaz de usuario de AWS ParallelCluster

La AWS ParallelCluster interfaz de usuario es una interfaz de usuario basada en la web que refleja la AWS ParallelCluster `pcluster` CLI y, al mismo tiempo, proporciona una experiencia similar a la de una consola. La AWS ParallelCluster interfaz de usuario se instala y se accede a ella en su Cuenta de AWS. Cuando la ejecutas, la AWS ParallelCluster interfaz de usuario accede a una instancia de la AWS ParallelCluster API alojada en Amazon API Gateway en tu Cuenta de AWS.

Note

Es posible que el asistente de AWS ParallelCluster interfaz de usuario no tenga opciones de interfaz de usuario para todas las funciones compatibles en la última AWS ParallelCluster versión compatible. Puede editar manualmente el archivo de configuración según sea necesario o utilizar la AWS ParallelCluster CLI.

En esta sección, le guiamos para la configuración y creación de un clúster mediante la interfaz de usuario de AWS ParallelCluster .

Requisitos previos:

- Acceso a una instancia de la AWS ParallelCluster interfaz de usuario en ejecución. Para obtener más información, consulte [Instalación de la interfaz de usuario de AWS ParallelCluster](#).

Configuración y creación de un clúster

1. En la vista de clústeres de la AWS ParallelCluster interfaz de usuario, selecciona Crear clúster, paso a paso.
2. En Clúster, Nombre, ingrese un nombre para su clúster.
3. Elija una VPC con una subred pública para su clúster y elija Siguiente.
4. En el Nodo principal, seleccione Añadir sesión de SSM y seleccione Siguiente.
5. En Colas, Recursos de computación, elija 1 para los Nodos estáticos.
6. En Tipo de instancia, elimine el tipo de instancia predeterminado seleccionado, elija t2.micro y elija Siguiente.
7. En Almacenamiento, seleccione Siguiente.
8. En Configuración del clúster, revise el YALM de la configuración del clúster y seleccione Ejecución de prueba para validarlo.
9. Seleccione Crear para crear el clúster en función de la configuración validada.
10. Transcurridos unos segundos, la AWS ParallelCluster interfaz de usuario te lleva automáticamente a los clústeres, donde puedes supervisar el estado de creación del clúster y los eventos de apilado.
11. Seleccione Detalles para ver los detalles del clúster, como la versión y el estado.
12. Elija Instancias para ver la lista de instancias de EC2 y el estado.
13. Selecciona Stack events para ver los eventos de la pila de clústeres y un AWS Management Console enlace a la CloudFormation pila que crea el clúster.
14. En Detalles, una vez finalizada la creación del clúster, elija Ver YAML para ver o descargar el archivo YAML de configuración del clúster.
15. Cuando se complete la creación del clúster, elija Shell para acceder al nodo principal del clúster.

Note

Al elegir Shell, AWS ParallelCluster abre una sesión de Amazon EC2 Systems Manager y añade `ssm-user` una `/etc/sudoers` a. Para obtener más información, consulte [Activar o desactivar los permisos administrativos de la cuenta `ssm-user`](#) en la Guía del usuario de Amazon EC2 Systems Manager.

16. Si quiere eliminarlo, en la vista Clústeres, seleccione el clúster y elija Acciones y Eliminar clúster.

Conectarse a un clúster

Al usar AWS ParallelCluster, puede conectarse al nodo principal del clúster para ejecutar trabajos, ver los resultados, administrar a usuarios y monitorear el estado del clúster y del trabajo. Conéctese a la instancia del nodo principal del clúster mediante los siguientes métodos:

- Inicie sesión mediante ssh con un [par de claves](#). Especifique la clave privada en [HeadNode/KeyName](#) en la configuración del clúster. Para obtener más información, consulte [Conexión a la instancia de Linux mediante un cliente SSH](#) en la Guía del usuario de instancias de Linux de Amazon EC2.
- Inicie sesión mediante el comando de la interfaz de la línea de comandos (CLI) `pcluster ssh`. Especifique la clave privada en [HeadNode/KeyName](#) de la configuración del clúster. Para obtener más información, consulte [pcluster ssh](#).
- Conéctese al nodo principal del clúster mediante una sesión SSM. Debe agregar la política administrada por AmazonSSManagedInstanceCore a [HeadNode/AdditionalIamPolicies](#) en la configuración del clúster para conectarse mediante una sesión SSM. Para obtener más información, consulte [SSM Session Manager](#) en la Guía del usuario de SSM.
- Conéctese al nodo principal del clúster mediante NICE DCV. Para obtener más información, consulte [Conexión al nodo principal a través de NICE DCV](#).
- Cuando utiliza la interfaz de usuario de AWS ParallelCluster, también puede conectarse al nodo principal del clúster mediante el comando EC2 Connect que proporciona la interfaz de usuario.

Acceso de varios usuarios a los clústeres

Aprenda a implementar y administrar el acceso de varios usuarios a un solo clúster.

En este tema, un usuario de AWS ParallelCluster hace referencia a un usuario del sistema para las instancias de computación. Un ejemplo es un `ec2-user` de una instancia de EC2 de AWS.

La compatibilidad del acceso multiusuario de AWS ParallelCluster está disponible en todas las Regiones de AWS en las que AWS ParallelCluster está disponible actualmente. Funciona con otros Servicios de AWS, como [Amazon FSx for Lustre](#) y [Amazon Elastic File System](#).

Puede usar un [AWS Directory Service for Microsoft Active Directory](#) o [Simple AD](#) para administrar el acceso al clúster. Asegúrese de comprobar la [disponibilidad de Región de AWS](#) para estos servicios. Para configurar un clúster, especifique una [AWS ParallelCluster DirectoryService](#) configuración.

AWS Directory Servicelos directorios se pueden conectar a varios clústeres. Esto permite una administración centralizada de las identidades en varios entornos y una experiencia de inicio de sesión unificada.

Si utiliza AWS Directory Service para el acceso de varios usuarios de AWS ParallelCluster, puede iniciar sesión en el clúster con las credenciales de usuario definidas en el directorio. Estas credenciales constan de un nombre de usuario y una contraseña. Tras iniciar sesión en el clúster por primera vez, se genera automáticamente una clave SSH de usuario. Puede utilizarla para iniciar sesión sin contraseña.

Puede crear, eliminar y modificar los usuarios o grupos de un clúster después de implementar el servicio de directorio. Con AWS Directory Service, puede hacerlo en la AWS Management Console o mediante la herramienta Usuarios y equipos de Active Directory. Se puede acceder a esta herramienta desde cualquier instancia de EC2 que esté unida a su Active Directory. Para obtener más información, consulte [Instalar las herramientas de administración de Active Directory](#).

Si piensa utilizar AWS ParallelCluster en una sola subred sin acceso a Internet, consulte [AWS ParallelCluster en una subred individual sin acceso a Internet](#) para conocer los requisitos adicionales.

Temas

- [Creación de un Active Directory](#)
- [Creación de un clúster con un dominio de AD](#)
- [Inicio de sesión en un clúster integrado con un dominio de AD](#)
- [Ejecución de trabajos de MPI](#)
- [Ejemplo de configuraciones de clústeres de AWS Managed Microsoft AD sobre LDAP\(S\)](#)

Creación de un Active Directory

Asegúrese de crear un Active Directory (AD) antes de crear el clúster. Para obtener información sobre cómo elegir el tipo de Active Directory para el clúster, consulte [Cuál elegir](#) en la Guía de administración de AWS Directory Service.

Si el directorio está vacío, añada usuarios con nombres de usuario y contraseñas. Para obtener más información, consulte la documentación específica de [AWS Directory Service for Microsoft Active Directory](#) o [Simple AD](#).

Note

AWS ParallelCluster requiere que todos los directorios de usuarios de Active Directory estén en el directorio `/home/$user`.

Creación de un clúster con un dominio de AD

Warning

En esta sección introductoria se describe cómo configurar AWS ParallelCluster un servidor Active Directory (AD) administrado mediante el Protocolo ligero de acceso a directorios (LDAP). El LDAP no es un protocolo seguro. Para los sistemas de producción, recomendamos encarecidamente el uso de certificados TLS (LDAPS), tal y como se describe en la siguiente sección [Ejemplo de configuraciones de clústeres de AWS Managed Microsoft AD sobre LDAP\(S\)](#).

Configure el clúster para que se integre con un directorio especificando la información pertinente en la sección `DirectoryService` del archivo de configuración del clúster. Para obtener más información, consulte la sección de configuración de [DirectoryService](#).

Puede usar el siguiente ejemplo para integrar el clúster con un AWS Managed Microsoft AD en el Protocolo ligero de acceso a directorios (LDAP).

Definiciones específicas que se requieren para una configuración de AWS Managed Microsoft AD basada en LDAP:

- El parámetro `ldap_auth_disable_tls_never_use_in_production` debe estar establecido como `True` en [DirectoryService/AdditionalSssdConfigs](#).
- Puede especificar los nombres de host del controlador o las direcciones IP para [DirectoryService/DomainAddr](#).
- La sintaxis [DirectoryService/DomainReadOnlyUser](#) debe ser la siguiente:

```
cn=ReadOnly,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
```

Obtenga los datos de configuración de AWS Managed Microsoft AD:


```
$ aws ds describe-directories --directory-id "d-abcdef01234567890"
```

```
{
  "DirectoryDescriptions": [
    {
      "DirectoryId": "d-abcdef01234567890",
      "Name": "corp.example.com",
      "DnsIpAddrs": [
        "203.0.113.225",
        "192.0.2.254"
      ],
      "VpcSettings": {
        "VpcId": "vpc-021345abcdef6789",
        "SubnetIds": [
          "subnet-1234567890abcdef0",
          "subnet-abcdef01234567890"
        ],
        "AvailabilityZones": [
          "region-idb",
          "region-idd"
        ]
      }
    }
  ]
}
```

Configuración de clúster para un AWS Managed Microsoft AD:

```
Region: region-id
Image:
  Os: alinux2
HeadNode:
  InstanceType: t2.micro
Networking:
  SubnetId: subnet-1234567890abcdef0
Ssh:
  KeyName: pcluster
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue1
  ComputeResources:
```

```
- Name: t2micro
  InstanceType: t2.micro
  MinCount: 1
  MaxCount: 10
Networking:
  SubnetIds:
    - subnet-abcdef01234567890
DirectoryService:
  DomainName: dc=corp,dc=example,dc=com
  DomainAddr: ldap://203.0.113.225,ldap://192.0.2.254
  PasswordSecretArn: arn:aws:secretsmanager:region-
id:123456789012:secret:MicrosoftAD.Admin.Password-1234
  DomainReadOnlyUser: cn=ReadOnly,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
  AdditionalSssdConfigs:
    ldap_auth_disable_tls_never_use_in_production: True
```

Para usar esta configuración para un Simple AD, cambie el valor de la propiedad **DomainReadOnlyUser** en la sección **DirectoryService**:

```
DirectoryService:
  DomainName: dc=corp,dc=example,dc=com
  DomainAddr: ldap://203.0.113.225,ldap://192.0.2.254
  PasswordSecretArn: arn:aws:secretsmanager:region-
id:123456789012:secret:SimpleAD.Admin.Password-1234
  DomainReadOnlyUser: cn=ReadOnlyUser,cn=Users,dc=corp,dc=example,dc=com
  AdditionalSssdConfigs:
    ldap_auth_disable_tls_never_use_in_production: True
```

Consideraciones:

- Le recomendamos que utilice LDAP sobre TLS/SSL (o LDAPS) en lugar de utilizar LDAP únicamente. El TLS/SSL garantiza que la conexión esté cifrada.
- El valor de la propiedad [DirectoryService/DomainAddr](#) coincide con las entradas de la lista `DnsIpAddrs` de la salida `describe-directories`.
- Se recomienda que el clúster utilice subredes que estén ubicadas en la misma zona de disponibilidad a la que apunte [DirectoryService/DomainAddr](#). Si utiliza una [configuración personalizada del Protocolo de configuración dinámica de host \(DHCP\)](#) que se recomienda para las VPC de directorio, y sus subredes no están ubicadas en la zona de disponibilidad de [DirectoryService/DomainAddr](#), es posible que se produzca tráfico cruzado entre las zonas

de disponibilidad. No es necesario usar configuraciones de DHCP personalizadas para usar la característica de integración de AD multiusuario.

- El valor de la propiedad [DirectoryService/DomainReadOnlyUser](#) especifica un usuario que debe crearse en el directorio. Este usuario no se crea de forma predeterminada. Le recomendamos que no dé permiso a este usuario para modificar los datos del directorio.
- El valor de la propiedad [DirectoryService/PasswordSecretArn](#) apunta a un secreto de AWS Secrets Manager que contiene la contraseña del usuario que especificó para la propiedad [DirectoryService/DomainReadOnlyUser](#). Si la contraseña de este usuario cambia, actualice el valor secreto y actualice el clúster. Para actualizar el clúster al nuevo valor secreto, debe detener la flota de procesamiento con el comando `pcluster update-compute-fleet`. Si ha configurado el clúster para usar [LoginNodes](#), detenga [LoginNodes/Pool](#)s y actualice el clúster después de establecer [LoginNodes/Pool/Count](#) en 0. A continuación, ejecute el siguiente comando desde el nodo principal del clúster.

```
sudo /opt/parallelcluster/scripts/directory_service/  
update_directory_service_password.sh
```

Para ver otro ejemplo, consulte también [Integración de Active Directory](#).

Inicio de sesión en un clúster integrado con un dominio de AD

Si ha activado la característica de integración de dominios de Active Directory (AD), la autenticación mediante contraseña se habilita en el nodo principal del clúster. El directorio principal de un usuario de AD se crea la primera vez que un usuario inicia sesión en el nodo principal o la primera vez que un `sudo-user` cambia al usuario de AD en el nodo principal.

La autenticación por contraseña no está habilitada para los nodos de computación del clúster. Los usuarios de AD deben iniciar sesión en los nodos de computación con claves SSH.

De forma predeterminada, las claves SSH se configuran en el directorio de usuarios de AD `/${HOME}/.ssh` al iniciar sesión por SSH por primera vez en el nodo principal.

Este comportamiento se puede deshabilitar configurando la propiedad booleana

[DirectoryService/GenerateSshKeysForUsers](#) como `false` en la configuración del clúster.

De forma predeterminada, [DirectoryService/GenerateSshKeysForUsers](#) está establecido en `true`.

Si una aplicación de AWS ParallelCluster requiere SSH sin contraseña entre los nodos del clúster, asegúrese de que las claves SSH estén configuradas correctamente en el directorio principal del usuario.

Las contraseñas de AWS Managed Microsoft AD caducarán al cabo de 42 días. Para obtener más información, consulte [Administración de las políticas de contraseñas de AWS Managed Microsoft AD](#) en la Guía de administración de AWS Directory Service. Si su contraseña caduca, debe restablecerla para restaurar el acceso al clúster. Para más información, consulte [Cómo restablecer una contraseña de usuario y contraseñas caducadas](#).

Note

Si la característica de integración de AD no funciona como se esperaba, los registros del SSSD pueden proporcionar información de diagnóstico útil para solucionar el problema. Estos registros se encuentran en el directorio `/var/log/sss` de los nodos del clúster. De forma predeterminada, también se almacenan en el grupo de CloudWatch registros de Amazon de un clúster.

Para más información, consulte [Solución de problemas de integración multiusuario con Active Directory](#).

Ejecución de trabajos de MPI

Como se sugiere en SchedMD, arranque los trabajos de MPI utilizando Slurm como método de arranque de MPI. Para obtener más información, consulte la [documentación oficial de Slurm](#) o la documentación oficial de su biblioteca MPI.

Por ejemplo, en la [documentación oficial de IntelMPI](#), aprenderá que, al ejecutar un trabajo de StarCCM, debe configurar Slurm como orquestador de procesos exportando la variable de entorno `I_MPI_HYDRA_BOOTSTRAP=slurm`.

Note

Problema conocido

En el caso de que su aplicación MPI utilice SSH como mecanismo para generar trabajos de MPI, es posible que se produzca un [error conocido en Slurm](#) que provoque una resolución incorrecta del nombre de usuario del directorio como “nadie”.

Configure su aplicación para que utilice Slurm como método de arranque de MPI o consulte [Problemas conocidos relacionados con la resolución del nombre de usuario](#) en la sección de solución de problemas para obtener más detalles y posibles soluciones alternativas.

Ejemplo de configuraciones de clústeres de AWS Managed Microsoft AD sobre LDAP(S)

AWS ParallelCluster admite el acceso de varios usuarios mediante la integración con un AWS Directory Service sobre un Protocolo ligero de acceso a directorios (LDAP) o LDAP sobre TLS/SSL (LDAPS).

En los siguientes ejemplos se muestra cómo crear configuraciones de clúster para integrarlas con un AWS Managed Microsoft AD sobre LDAP(S).

AWS Managed Microsoft AD sobre LDAPS con verificación de certificados

Puede usar este ejemplo para integrar su clúster con un AWS Managed Microsoft AD sobre LDAPS, con verificación de certificados.

Definiciones específicas para una configuración de AWS Managed Microsoft AD sobre LDAPS con certificados:

- [DirectoryService/LdapTlsReqCert](#) debe establecerse en `hard` (predeterminado) para LDAPS con verificación de certificados.
- [DirectoryService/LdapTlsCaCert](#) debe especificar la ruta de acceso a su certificado de autoridad (CA).

El certificado de CA es un paquete de certificados que contiene los certificados de toda la cadena de CA que emitió los certificados para los controladores de dominio de AD.

Los certificados y el certificado de CA deben estar instalados en los nodos del clúster.

- Los nombres de host de los controladores deben especificarse para [DirectoryService/DomainAddr](#), no para las direcciones IP.
- La sintaxis [DirectoryService/DomainReadOnlyUser](#) debe ser la siguiente:

```
cn=ReadOnly,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
```

Ejemplo de archivo de configuración de clúster para usar AD sobre LDAPS:

```
Region: region-id
Image:
  Os: alinux2
HeadNode:
  InstanceType: t2.micro
  Networking:
    SubnetId: subnet-1234567890abcdef0
  Ssh:
    KeyName: pcluster
  Iam:
    AdditionalIamPolicies:
      - Policy: arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
  CustomActions:
    OnNodeConfigured:
      Script: s3://aws-parallelcluster/scripts/pcluster-dub-msad-ldaps.post.sh
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue1
      ComputeResources:
        - Name: t2micro
          InstanceType: t2.micro
          MinCount: 1
          MaxCount: 10
      Networking:
        SubnetIds:
          - subnet-abcdef01234567890
      Iam:
        AdditionalIamPolicies:
          - Policy: arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
      CustomActions:
        OnNodeConfigured:
          Script: s3://aws-parallelcluster-pcluster/scripts/pcluster-dub-msad-
ldaps.post.sh
DirectoryService:
  DomainName: dc=corp,dc=example,dc=com
  DomainAddr: ldaps://win-abcdef01234567890.corp.example.com,ldaps://win-
abcdef01234567890.corp.example.com
  PasswordSecretArn: arn:aws:secretsmanager:region-
id:123456789012:secret:MicrosoftAD.Admin.Password-1234
  DomainReadOnlyUser: cn=ReadOnly,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
  LdapTlsCaCert: /etc/openldap/cacerts/corp.example.com.bundleca.cer
```

```
LdapTlsReqCert: hard
```

Agregue certificados y configure los controladores de dominio en el script posterior a la instalación:

```
*#!/bin/bash*
set -e

AD_CERTIFICATE_S3_URI="s3://corp.example.com/bundle/corp.example.com.bundleca.cer"
AD_CERTIFICATE_LOCAL="/etc/openssl/cacerts/corp.example.com.bundleca.cer"

AD_HOSTNAME_1="win-abcdef01234567890.corp.example.com"
AD_IP_1="192.0.2.254"

AD_HOSTNAME_2="win-abcdef01234567890.corp.example.com"
AD_IP_2="203.0.113.225"

# Download CA certificate
mkdir -p $(dirname "${AD_CERTIFICATE_LOCAL}")
aws s3 cp "${AD_CERTIFICATE_S3_URI}" "${AD_CERTIFICATE_LOCAL}"
chmod 644 "${AD_CERTIFICATE_LOCAL}"

# Configure domain controllers reachability
echo "${AD_IP_1} ${AD_HOSTNAME_1}" >> /etc/hosts
echo "${AD_IP_2} ${AD_HOSTNAME_2}" >> /etc/hosts
```

Puede recuperar los nombres de host de los controladores de dominio de las instancias unidas al dominio, como se muestra en los siguientes ejemplos.

Desde una instancia de Windows

```
$ nslookup 192.0.2.254
```

```
Server: corp.example.com
Address: 192.0.2.254

Name: win-abcdef01234567890.corp.example.com
Address: 192.0.2.254
```

Desde una instancia de Linux

```
$ nslookup 192.0.2.254
```

```
192.0.2.254.in-addr.arpa  name = corp.example.com
192.0.2.254.in-addr.arpa  name = win-abcdef01234567890.corp.example.com
```

AWS Managed Microsoft AD sobre LDAPS sin verificación de certificados

Puede usar este ejemplo para integrar su clúster con un AWS Managed Microsoft AD sobre LDAPS, sin verificación de certificados.

Definiciones específicas para una configuración de AWS Managed Microsoft AD sobre LDAPS sin verificación de certificados:

- [DirectoryService/LdapTlsReqCert](#) debe establecerse en `never`.
- Se pueden especificar los nombres de host del controlador o las direcciones IP para [DirectoryService/DomainAddr](#).
- La sintaxis [DirectoryService/DomainReadOnlyUser](#) debe ser la siguiente:

```
cn=ReadOnly,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
```

Ejemplo de archivo de configuración de clúster para usar AWS Managed Microsoft AD sobre LDAPS sin verificación de certificados:

```
Region: region-id
Image:
  Os: alinux2
HeadNode:
  InstanceType: t2.micro
  Networking:
    SubnetId: subnet-1234567890abcdef0
  Ssh:
    KeyName: pcluster
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue1
      ComputeResources:
        - Name: t2micro
          InstanceType: t2.micro
          MinCount: 1
          MaxCount: 10
      Networking:
```



```
SubnetIds:
  - subnet-abcdef01234567890
DirectoryService:
  DomainName: dc=corp,dc=example,dc=com
  DomainAddr: ldaps://203.0.113.225,ldaps://192.0.2.254
  PasswordSecretArn: arn:aws:secretsmanager:region-
id:123456789012:secret:MicrosoftAD.Admin.Password-1234
  DomainReadOnlyUser: cn=ReadOnly,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
  LdapTlsReqCert: never
```

Prácticas recomendadas

Prácticas recomendadas: selección del tipo de instancia del nodo principal

Aunque el nodo principal no ejecuta ningún trabajo, sus funciones y su tamaño son cruciales para el rendimiento general del clúster. Al elegir el tipo de instancia que se utilizará para el nodo principal, tenga en cuenta las siguientes características:

Tamaño del clúster: el nodo principal organiza la lógica de escalado del clúster y es responsable de adjuntar los nuevos nodos al programador. Para escalar y reducir verticalmente un clúster que tenga un gran número de nodos, proporcione al nodo principal capacidad de computación adicional.

Sistemas de archivos compartidos: cuando utilice sistemas de archivos compartidos, elija un tipo de instancia con suficiente ancho de banda de la red y suficiente ancho de banda de Amazon EBS para administrar sus flujos de trabajo. Asegúrese de que el nodo principal pueda exponer suficientes directorios de servidores NFS para el clúster y administrar los artefactos que deben compartirse entre los nodos de computación y el nodo principal.

Prácticas recomendadas: rendimiento de la red

El rendimiento de la red es fundamental para las aplicaciones de computación de alto rendimiento (HPC). Sin un rendimiento de red fiable, estas aplicaciones no pueden funcionar según lo esperado. Para optimizar el rendimiento de la red, tenga en cuenta las siguientes prácticas recomendadas.

- **Grupo de ubicación:** si utiliza Slurm, considere la posibilidad de configurar cada cola de Slurm para que utilice un grupo con ubicación en clúster. Un grupo con ubicación en clúster es una agrupación lógica de instancias en una misma zona de disponibilidad. Para obtener más información, consulte [los grupos de ubicación](#) en la Guía del usuario de Amazon EC2. Puede especificar un [PlacementGroup](#) en la sección [Networking](#) de la cola; cada recurso de computación se asigna al grupo de ubicación de la cola. Al especificar un

[PlacementGroup](#) en la sección [Networking](#) del recurso de computación, se asigna ese recurso de computación específico a ese grupo de ubicación. La especificación del grupo de ubicación del recurso de computación anula la especificación de la cola del recurso de computación. Para obtener más información, consulte [SlurmQueues/Networking/PlacementGroup](#) y [SlurmQueues/ComputeResources/Networking/PlacementGroup](#).

```
Networking:
  PlacementGroup:
    Enabled: true
    Id: your-placement-group-name
```

También puede solicitar que AWS ParallelCluster cree un grupo de colocación para usted.

```
Networking:
  PlacementGroup:
    Enabled: true
```

A partir de AWS ParallelCluster la versión 3.3.0, se modifica la creación y administración de los grupos de colocación. Al especificar el grupo de ubicación que se va a habilitar, sin un name o Id, en la cola, a cada recurso de computación se le asigna su propio grupo de ubicación administrado, en lugar de un grupo administrado para toda la cola. Esto ayuda a reducir los errores por capacidad insuficiente. Si necesita tener un grupo de ubicación para toda la cola, puede usar un grupo de ubicación con nombre.

Se ha agregado [SlurmQueues/Networking/PlacementGroup/Name](#) como alternativa preferida a [SlurmQueues/Networking/PlacementGroup/Id](#).

Para obtener más información, consulte [Networking](#).

- **Redes mejoradas:** considere la posibilidad de elegir un tipo de instancia que admita redes mejoradas. Esta recomendación se aplica a todas las [instancias de la generación actual](#). Para obtener más información, consulte [redes mejoradas en Linux](#) en la Guía del usuario de Amazon EC2.
- **Elastic Fabric Adapter:** para admitir altos niveles de comunicación escalable de instancia a instancia, considere la posibilidad de elegir interfaces de red EFA para su red. El hardware de derivación del sistema operativo personalizado de la EFA mejora las comunicaciones de instancia a instancia con la elasticidad y flexibilidad bajo demanda de la Nube de AWS. Puede configurar cada cola de Slurm [ComputeResource](#) para que utilice [Efa](#). Para obtener más información sobre el uso de EFA con AWS ParallelCluster, consulte. [Elastic Fabric Adapter](#)

```
ComputeResources:
  - Name: your-compute-resource-name
    Efa:
      Enabled: true
```

Para obtener más información sobre EFA, consulte [Elastic Fabric Adapter](#) en la Guía del usuario para instancias de Linux de Amazon EC2.

- Ancho de banda de la instancia: el ancho de banda se escala con el tamaño de la instancia. Para obtener información sobre los distintos tipos de [instancias](#), consulte [Instancias optimizadas para Amazon EBS](#) y [tipos de volumen de Amazon EBS](#) en la Guía del usuario de Amazon EC2.

Prácticas recomendadas: alertas de presupuesto

Para gestionar los costes de los recursos AWS ParallelCluster, le recomendamos que utilice AWS Budgets acciones para crear un presupuesto. También puede crear alertas de umbrales presupuestarios definidos para AWS los recursos seleccionados. Para obtener más información, consulte [Configuring a budget action](#) en la Guía del usuario de AWS Budgets . Del mismo modo, también puedes usar Amazon CloudWatch para crear una alarma de facturación. Para obtener más información, consulte [Creación de una alarma de facturación para monitorear los cargos estimados de AWS](#).

Mejores prácticas: mover un clúster a una nueva versión AWS

ParallelCluster secundaria o a una versión de parche

Actualmente, cada versión AWS ParallelCluster secundaria es autónoma junto con su `pcluster` CLI. Para mover un clúster a una nueva versión secundaria o de parche, debe volver a crear el clúster mediante la CLI de la nueva versión.

Para optimizar el proceso de traslado de un clúster a una nueva versión secundaria o de parche, le recomendamos que haga lo siguiente:

- Guarde los datos personales en volúmenes externos que se crean fuera del clúster, como Amazon EFS y FSx para Lustre. De este modo, podrá mover fácilmente los datos de un clúster a otro en el futuro.
- Cree sistemas de almacenamiento compartido con los siguientes tipos. Puede crear estos sistemas mediante la tecla AWS CLI o AWS Management Console.

- [SharedStorage](#) / [EbsSettings](#) / [VolumeId](#)
- [SharedStorage](#) / [EfsSettings](#) / [FileSystemId](#)
- [SharedStorage](#) / [FsxLustreSettings](#) / [FileSystemId](#)

Defina un sistema de archivos o un volumen en una configuración de clúster como sistema de archivos o volumen existente. De esta forma, se conservan al eliminar el clúster y se pueden asociar a un clúster nuevo.

Le recomendamos que utilice sistemas de archivos de Amazon EFS o FSx para Lustre. Ambos sistemas se pueden conectar a varios clústeres al mismo tiempo. Además, puede asociar cualquiera de estos sistemas a un clúster nuevo antes de eliminar el clúster existente.

- Use las [acciones de arranque personalizadas](#) para personalizar sus instancias en lugar de usar una AMI personalizada. Si, por el contrario, utiliza una AMI personalizada, tendrá que eliminar y volver a crear esa AMI para cada versión nueva.
- Se recomienda aplicar las recomendaciones anteriores en la secuencia siguiente:
 1. Actualice la configuración del clúster existente para utilizar las definiciones de sistemas de archivos existentes.
 2. Compruebe la versión de `pc1uster` y actualícela si es necesario.
 3. Cree y pruebe el nuevo clúster. Al probar el nuevo clúster, compruebe lo siguiente:
 - Asegúrese de que sus datos estén disponibles en el clúster nuevo.
 - Asegúrese de que la aplicación funcione en el clúster nuevo.
 4. Cuando haya probado por completo el clúster nuevo, esté en funcionamiento y ya no necesite el clúster existente, elimínelo.

Transición de AWS ParallelCluster 2.x a 3.x

Acciones de arranque personalizadas

Con AWS ParallelCluster 3, puede especificar diferentes scripts de acciones de arranque personalizados para el nodo principal y los nodos de computación utilizando los parámetros `OnNodeStart` (`pre_install` en la versión 2 de AWS ParallelCluster) y `OnNodeConfigured` (`post_install` en la versión 2 de AWS ParallelCluster) en las secciones [HeadNode](#) y [Scheduling/SlurmQueues](#). Para obtener más información, consulte [Acciones de arranque personalizadas](#).

Los scripts de acciones de arranque personalizados que se han desarrollado para AWS ParallelCluster 2 deben adaptarse para su uso en AWS ParallelCluster 3:

- No recomendamos usar `/etc/parallelcluster/cfnconfig` ni `cfn_node_type` para diferenciar entre nodos principales y de computación. En su lugar, le recomendamos que especifique dos scripts diferentes en [HeadNode](#) y [Scheduling/SlurmQueues](#).
- Si prefiere seguir cargando `/etc/parallelcluster/cfnconfig` para usarlo en su script de acciones de arranque, tenga en cuenta que el valor de `cfn_node_type` ha cambiado de “MasterServer” a “HeadNode” (consulte: [Lenguaje inclusivo](#)).
- En AWS ParallelCluster 2, el primer argumento de entrada para los scripts de acciones de arranque era la URL S3 del script y estaba reservado. En AWS ParallelCluster 3, solo los argumentos configurados en la configuración se pasan a los scripts.

Warning

Oficialmente, no se admite el uso de variables internas proporcionadas a través del archivo `/etc/parallelcluster/cfnconfig`. Es posible que este archivo se elimine como parte de una versión futura.

AWS ParallelCluster 2.x y 3.x utilizan una sintaxis de archivo de configuración diferente

La configuración de AWS ParallelCluster 3.x usa la sintaxis YAML. La referencia completa se encuentra en [Archivos de configuración](#).

Además de requerir un formato de archivo YAML, en AWS ParallelCluster 3.x se han actualizado varias secciones de configuración, ajustes y valores de parámetros. En esta sección, analizamos los cambios clave en la configuración de AWS ParallelCluster junto con ejemplos paralelos que ilustran estas diferencias en cada versión de AWS ParallelCluster.

Ejemplo de configuración de varias colas del programador con el hiperprocesamiento habilitado y deshabilitado

AWS ParallelCluster 2:

```
[cluster default]
queue_settings = ht-enabled, ht-disabled
```

```

...

[queue ht-enabled]
compute_resource_settings = ht-enabled-i1
disable_hyperthreading = false

[queue ht-disabled]
compute_resource_settings = ht-disabled-i1
disable_hyperthreading = true

[compute_resource ht-enabled-i1]
instance_type = c5n.18xlarge
[compute_resource ht-disabled-i1]
instance_type = c5.xlarge

```

AWS ParallelCluster 3:

```

...
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: ht-enabled
      Networking:
        SubnetIds:
          - compute_subnet_id
      ComputeResources:
        - Name: ht-enabled-i1
          DisableSimultaneousMultithreading: true
          InstanceType: c5n.18xlarge
    - Name: ht-disabled
      Networking:
        SubnetIds:
          - compute_subnet_id
      ComputeResources:
        - Name: ht-disabled-i1
          DisableSimultaneousMultithreading: false
          InstanceType: c5.xlarge

```

Ejemplo de nueva configuración de sistema de archivos FSx for Lustre

AWS ParallelCluster 2:

```
[cluster default]
```

```

fsx_settings = fsx
...

[fsx fsx]
shared_dir = /shared-fsx
storage_capacity = 1200
imported_file_chunk_size = 1024
import_path = s3://bucket
export_path = s3://bucket/export_dir
weekly_maintenance_start_time = 3:02:30
deployment_type = PERSISTENT_1
data_compression_type = LZ4

```

AWS ParallelCluster 3:

```

...
SharedStorage:
- Name: fsx
  MountDir: /shared-fsx
  StorageType: FsxLustre
  FsxLustreSettings:
    StorageCapacity: 1200
    ImportedFileChunkSize: 1024
    ImportPath: s3://bucket
    ExportPath: s3://bucket/export_dir
    WeeklyMaintenanceStartTime: "3:02:30"
    DeploymentType: PERSISTENT_1
    DataCompressionType: LZ4

```

Ejemplo de una configuración de clúster que monta un sistema de archivos FSx for Lustre existente

AWS ParallelCluster 2:

```

[cluster default]
fsx_settings = fsx
...

[fsx fsx]
shared_dir = /shared-fsx
fsx_fs_id = fsx_fs_id

```

AWS ParallelCluster 3:

```

...
SharedStorage:
  - Name: fsx
    MountDir: /shared-fsx
    StorageType: FsxLustre
    FsxLustreSettings:
      FileSystemId: fsx_fs_id

```

Ejemplo de un clúster con la pila de software de especificación de plataforma Intel HPC

AWS ParallelCluster 2:

```

[cluster default]
enable_intel_hpc_platform = true
...

```

AWS ParallelCluster 3:

```

...
AdditionalPackages:
  IntelSoftware:
    IntelHpcPlatform: true

```

Notas:

- La instalación del software de especificación de plataforma HPC Intel está sujeta a los términos y condiciones del [acuerdo de licencia de usuario final de Intel](#) aplicable.

Ejemplo de configuraciones de IAM personalizadas, que incluyen: el perfil de la instancia, el rol de la instancia, las políticas adicionales para las instancias y el rol de las funciones de lambda asociadas al clúster

AWS ParallelCluster 2:

```

[cluster default]
additional_iam_policies = arn:aws:iam::aws:policy/
AmazonS3ReadOnlyAccess,arn:aws:iam::aws:policy/AmazonDynamoDBReadOnlyAccess
ec2_iam_role = ec2_iam_role
iam_lambda_role = lambda_iam_role
...

```


AWS ParallelCluster 3:

```

...
Iam:
  Roles:
    CustomLambdaResources: lambda_iam_role
HeadNode:
  ...
  Iam:
    InstanceRole: ec2_iam_role
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue1
      ...
      Iam:
        InstanceProfile: iam_instance_profile
    - Name: queue2
      ...
      Iam:
        AdditionalIamPolicies:
          - Policy: arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
          - Policy: arn:aws:iam::aws:policy/AmazonDynamoDBReadOnlyAccess

```

Notas:

- En el caso de AWS ParallelCluster 2, la configuración de IAM se aplica a todas las instancias de un clúster y no se puede usar `additional_iam_policies` junto con `ec2_iam_role`.
- En el caso de AWS ParallelCluster 3, puede tener diferentes configuraciones de IAM para los nodos principales y de computación e incluso especificar diferentes configuraciones de IAM para cada cola de computación.
- En el caso de AWS ParallelCluster 3, puede utilizar un perfil de instancia de IAM como alternativa a un rol de IAM. `InstanceProfile`, `InstanceRole` o `AdditionalIamPolicies` no se pueden configurar juntos.

Ejemplo de acciones de arranque personalizadas

AWS ParallelCluster 2:

```

[cluster default]
s3_read_resource = arn:aws:s3:::bucket_name/*

```

```
pre_install = s3://bucket_name/scripts/pre_install.sh
pre_install_args = 'R curl wget'
post_install = s3://bucket_name/scripts/post_install.sh
post_install_args = "R curl wget"
...
```

AWS ParallelCluster 3:

```
...
HeadNode:
  ...
  CustomActions:
    OnNodeStart:
      Script: s3://bucket_name/scripts/pre_install.sh
      Args:
        - R
        - curl
        - wget
    OnNodeConfigured:
      Script: s3://bucket_name/scripts/post_install.sh
      Args: ['R', 'curl', 'wget']
  Iam:
    S3Access:
      - BucketName: bucket_name
  Scheduling:
    Scheduler: slurm
    SlurmQueues:
      - Name: queue1
      ...
    CustomActions:
      OnNodeStart:
        Script: s3://bucket_name/scripts/pre_install.sh
        Args: ['R', 'curl', 'wget']
      OnNodeConfigured:
        Script: s3://bucket_name/scripts/post_install.sh
        Args: ['R', 'curl', 'wget']
  Iam:
    S3Access:
      - BucketName: bucket_name
```

Ejemplo de un clúster con acceso de lectura y escritura a los recursos del bucket de S3

AWS ParallelCluster 2:

```
[cluster default]
s3_read_resource = arn:aws:s3:::bucket/read_only/*
s3_read_write_resource = arn:aws:s3:::bucket/read_and_write/*
...
```

AWS ParallelCluster 3:

```
...
HeadNode:
  ...
  Iam:
    S3Access:
      - BucketName: bucket_name
        KeyName: read_only/
        EnableWriteAccess: False
      - BucketName: bucket_name
        KeyName: read_and_write/
        EnableWriteAccess: True
  Scheduling:
    Scheduler: slurm
    SlurmQueues:
      - Name: queue1
      ...
    Iam:
      S3Access:
        - BucketName: bucket_name
          KeyName: read_only/
          EnableWriteAccess: False
        - BucketName: bucket_name
          KeyName: read_and_write/
          EnableWriteAccess: True
```

Lenguaje inclusivo

AWS ParallelCluster 3 usa los términos “nodo principal” en los lugares donde se usó “maestro” en AWS ParallelCluster 2. Estas incluyen las siguientes:

- La variable exportada en el entorno de trabajo de AWS Batch ha cambiado de MASTER_IP a PCLUSTER_HEAD_NODE_IP.
- Todas las salidas de AWS CloudFormation han cambiado de Master* a HeadNode*.
- Todos los NodeType y las etiquetas han cambiado de Master a HeadNode.

Compatibilidad con el programador

AWS ParallelCluster 3.x no es compatible con los programadores Son of Grid Engine (SGE) y Torque.

Los comandos de AWS Batch `awsbhosts`, `awsbkill`, `awsbout`, `awsbqueues`, `awsbstat` y `awsbsub` se distribuyen como un paquete PyPI `aws-parallelcluster-awsbatch-cli` independiente. Este paquete lo instala AWS ParallelCluster en el nodo principal. Puede seguir utilizando estos comandos de AWS Batch desde el nodo principal del clúster. Sin embargo, si desea utilizar los comandos AWS Batch desde una ubicación que no sea el nodo principal, primero debe instalar el paquete PyPI `aws-parallelcluster-awsbatch-cli`.

AWS ParallelCluster CLI

Se ha cambiado la interfaz de la línea de comandos (CLI) de AWS ParallelCluster. La nueva sintaxis se describe en [Comandos de la CLI de AWS ParallelCluster](#). El formato de salida de la CLI es una cadena [JSON](#).

Configuración de un clúster nuevo

El comando `pcluster configure` incluye parámetros diferentes en AWS ParallelCluster 3 en comparación con AWS ParallelCluster 2. Para obtener más información, consulte [pcluster configure](#).

Tenga en cuenta también que la sintaxis del archivo de configuración ha cambiado desde AWS ParallelCluster 2. Para obtener una referencia completa de los ajustes de configuración del clúster, consulte [Configuración del clúster](#).

Creación de un clúster

El comando `pcluster create` de AWS ParallelCluster 2 se ha sustituido por el comando [pcluster create-cluster](#).

Tenga en cuenta que el comportamiento predeterminado en AWS ParallelCluster 2.x, sin la opción `-nw`, es esperar a que se creen los clústeres, mientras que el comando de AWS ParallelCluster 3.x realiza una devolución inmediata. El progreso de la creación del clúster se puede monitorear mediante [pcluster describe-cluster](#).

Un archivo de configuración de AWS ParallelCluster 3 contiene una única definición de clúster, por lo que el parámetro `-t` ya no es necesario.

A continuación, se muestra un ejemplo de archivo de configuración.

```
# AWS ParallelCluster v2
$ pcluster create \
  -r REGION \
  -c V2_CONFIG_FILE \
  -nw \
  -t CLUSTER_TEMPLATE \
  CLUSTER_NAME

# AWS ParallelCluster v3
$ pcluster create-cluster \
  --region REGION \
  --cluster-configuration V3_CONFIG_FILE \
  --cluster-name CLUSTER_NAME
```

Mostrar clústeres

El comando AWS ParallelCluster de `pcluster list 2.x` debe sustituirse por el comando [pcluster list-clusters](#).

Nota: Necesita la CLI de AWS ParallelCluster versión 2 para mostrar los clústeres creados con las versiones 2.x de AWS ParallelCluster. Consulte [Instalar AWS ParallelCluster en un entorno virtual \(recomendado\)](#) para descubrir cómo instalar varias versiones de AWS ParallelCluster usando entornos virtuales.

```
# AWS ParallelCluster v2
$ pcluster list -r REGION

# AWS ParallelCluster v3
$ pcluster list-clusters --region REGION
```

Inicio y detención de un clúster

Los comandos `pcluster stop` y AWS ParallelCluster de `pcluster start 2.x` deben sustituirse por los comandos [pcluster update-compute-fleet](#).

Inicio de una flota de computación:

```
# AWS ParallelCluster v2
$ pcluster start \
```

```
-r REGION \  
CLUSTER_NAME  
  
# AWS ParallelCluster v3 - Slurm fleets  
$ pcluster update-compute-fleet \  
  --region REGION \  
  --cluster-name CLUSTER_NAME \  
  --status START_REQUESTED  
  
# AWS ParallelCluster v3 - AWS Batch fleets  
$ pcluster update-compute-fleet \  
  --region REGION \  
  --cluster-name CLUSTER_NAME \  
  --status ENABLED
```

Detención de una flota de computación:

```
# AWS ParallelCluster v2  
$ pcluster stop \  
  -r REGION \  
  CLUSTER_NAME  
  
# AWS ParallelCluster v3 - Slurm fleets  
$ pcluster update-compute-fleet \  
  --region REGION \  
  --cluster-name CLUSTER_NAME \  
  --status STOP_REQUESTED  
  
# AWS ParallelCluster v3 - AWS Batch fleets  
$ pcluster update-compute-fleet \  
  --region REGION \  
  --cluster-name CLUSTER_NAME \  
  --status DISABLED
```

Conexión a un clúster

El comando AWS ParallelCluster de `pcluster ssh 2.x` tiene diferentes nombres de parámetros en AWS ParallelCluster 3.x. Consulte [pcluster ssh](#).

Conexión a un clúster:

```
# AWS ParallelCluster v2  
$ pcluster ssh \
```

```

-r REGION \
CLUSTER_NAME \
-i ~/.ssh/id_rsa

# AWS ParallelCluster v3
$ pcluster ssh \
  --region REGION \
  --cluster-name CLUSTER_NAME \
  -i ~/.ssh/id_rsa

```

Actualización de la configuración de IMDS

A partir de la versión 3.0.0, AWS ParallelCluster introdujo la compatibilidad para restringir el acceso al IMDS del nodo principal (y a las credenciales del perfil de la instancia) a un subconjunto de superusuarios, de forma predeterminada. Para obtener más información, consulte [Propiedades de Imds](#).

Regiones compatibles con AWS ParallelCluster

AWS ParallelCluster versión 3 está disponible en la siguiente Regiones de AWS:

Nombre de la región	Región
Este de EE. UU. (Ohio)	us-east-2
Este de EE. UU. (Norte de Virginia)	us-east-1
Oeste de EE. UU. (Norte de California)	us-west-1
Oeste de EE. UU. (Oregón)	us-west-2
África (Ciudad del Cabo)	af-south-1
Asia-Pacífico (Hong Kong)	ap-east-1
Asia-Pacífico (Bombay)	ap-south-1
Asia-Pacífico (Seúl)	ap-northeast-2
Asia-Pacífico (Singapur)	ap-southeast-1

Nombre de la región	Región
Asia-Pacífico (Sídney)	ap-southeast-2
Asia-Pacífico (Tokio)	ap-northeast-1
Canadá (centro)	ca-central-1
China (Beijing)	cn-north-1
China (Ningxia)	cn-northwest-1
Europe (Frankfurt)	eu-central-1
Europa (Irlanda)	eu-west-1
Europa (Londres)	eu-west-2
Europa (Milán)	eu-south-1
Europa (París)	eu-west-3
Europa (Estocolmo)	eu-north-1
Medio Oriente (Baréin)	me-south-1
América del Sur (São Paulo)	sa-east-1
AWS GovCloud (Este de EE. UU.)	us-gov-east-1
AWS GovCloud (Estados Unidos-Oeste)	us-gov-west-1
Israel (Tel Aviv)	il-central-1

Uso AWS ParallelCluster

Temas

- [IU AWS ParallelCluster](#)
- [Configuración de la VPC de AWS Lambda en AWS ParallelCluster](#)
- [AWS Identity and Access Management permisos en AWS ParallelCluster](#)
- [Configuraciones de red](#)
- [Nodos de inicio de sesión](#)
- [Acciones de arranque personalizadas](#)
- [Uso de Amazon S3](#)
- [Uso de instancias de spot](#)
- [Programadores compatibles con AWS ParallelCluster](#)
- [Almacenamiento compartido](#)
- [Recursos y etiquetado de AWS ParallelCluster](#)
- [Monitoreo de AWS ParallelCluster y registros](#)
- [AWS CloudFormation recurso personalizado](#)
- [Elastic Fabric Adapter](#)
- [Habilitar Intel MPI](#)
- [API de AWS ParallelCluster](#)
- [Conexión al nodo principal a través de NICE DCV](#)
- [Uso de pcluster update-cluster](#)
- [AWS ParallelCluster Personalización de AMI](#)
- [Inicio de instancias con ODCR \(reservas de capacidad bajo demanda\)](#)
- [Parcheo de AMI y sustitución de instancias de EC2](#)
- [Sistemas operativos](#)

IU AWS ParallelCluster

La interfaz de usuario AWS ParallelCluster es una interfaz de usuario basada en la web que sirve como panel para crear, monitorear y administrar clústeres. La interfaz de usuario de AWS

ParallelCluster se instala en la Cuenta de AWS, desde donde también puede acceder a ella. La IU AWS ParallelCluster se añade con AWS ParallelCluster versión 3.5.0.

Para instalar la interfaz de usuario AWS ParallelCluster y empezar, consulte [Instalación de la interfaz de usuario de AWS ParallelCluster](#) y [Configuración y creación de un clúster con la interfaz de usuario de AWS ParallelCluster](#).

The screenshot displays the AWS ParallelCluster user interface. At the top, the header shows the AWS ParallelCluster logo, the user's email (user@domain.com), and the region (eu-west-1). The main navigation menu on the left includes 'Clusters', 'Images', and 'Users'. The main content area is titled 'Clusters (2)' and contains a search bar and a table of clusters. The table has columns for Name, Status, and Version. Two clusters are listed: 'hpc-cluster-1' with a status of 'CREATE COMPLETE' and version '3.5.0', and 'hpc-cluster-2' with a status of 'DELETE IN PROGRESS' and version '3.5.0'. Below the table, the details for 'hpc-cluster-1' are shown, including tabs for 'Details', 'Instances', 'Storage', 'Job scheduling', and 'Stack events'. The 'Details' tab is active, showing properties such as Cluster configuration, SSH command, EC2 Instance Connect, Cluster status (CREATE COMPLETE), Compute fleet status (RUNNING), Version (3.5.0), Region (eu-west-1), Created time (March 10, 2023 at 09:39 (UTC+1:00)), and Latest update time (March 10, 2023 at 09:39 (UTC+1:00)).


La IU AWS ParallelCluster es compatible con las siguientes características:

- Muestra lo siguiente:
 - La lista de clústeres que has creado en Cuenta de AWS con AWS ParallelCluster.
 - El estado y los detalles disponibles de los clústeres de la lista.
 - Los eventos y los registros AWS ParallelCluster de CloudFormation se pueden utilizar para la supervisión.

- El estado de los trabajos que se están ejecutando en sus clústeres.
- La lista de imágenes personalizadas que puede usar para crear clústeres.
- La lista de imágenes oficiales que utiliza la interfaz de usuario para crear clústeres.
- La lista de usuarios que tienen acceso a la interfaz de usuario AWS ParallelCluster. Puede añadir y eliminar usuarios.
- Proporciona instrucciones paso a paso para crear y editar (actualizar) un clúster y seleccionar las funciones de clúster compatibles para añadirlas, editarlas o eliminarlas. Los campos de entrada inaccesibles no se pueden cambiar para la configuración del clúster que se está editando. Tiene la opción de realizar una validación de prueba de la configuración del clúster antes de la implementación del clúster.
- Incluye enlaces directos desde el shell para acceder al nodo principal en la vista de clústeres. Seleccione Añadir sesión de SSM en las instrucciones paso a paso para añadir el acceso directo al shell y la política Instancia principal gestionada por SSM en el nodo principal.

Tenga en cuenta lo siguiente cuando utilice la interfaz de usuario AWS ParallelCluster para crear y administrar sus clústeres:

- Solo puede crear y editar clústeres o crear imágenes con la misma versión de AWS ParallelCluster que se usó para crear la interfaz de usuario de AWS ParallelCluster. Las imágenes o los clústeres de las versiones anteriores solo se pueden ver. Si administra varias versiones de clústeres e imágenes, le recomendamos que cree una instancia de interfaz de usuario AWS ParallelCluster compatible con cada versión.
- La interfaz de usuario AWS ParallelCluster está diseñada para reflejar la funcionalidad de la CLI `pccluster`. Hay algunas diferencias. Si sigue las instrucciones paso a paso, estará utilizando todas las funciones compatibles. Antes de la implementación, tiene la opción de editar la configuración del clúster o de la imagen manualmente. Si lo hace, le recomendamos que valide la configuración seleccionando Ejecutar en seco para comprobar que las modificaciones son totalmente compatibles.

 Note

La IU AWS ParallelCluster no es compatible con AWS Batch.

Configuración de la VPC de AWS Lambda en AWS ParallelCluster

AWS ParallelCluster utiliza AWS Lambda para realizar operaciones durante el ciclo de vida del clúster. Una función de [AWS Lambda siempre se ejecuta en una VPC](#) propiedad del servicio de Lambda. La función de Lambda también se puede conectar a subredes privadas en una nube privada virtual (VPC) para acceder a los recursos privados.

Note

Las funciones Lambda no pueden conectarse directamente a una VPC con tenencia de instancias dedicada. Para conectarse a los recursos de una VPC dedicada, interconéctela con una segunda VPC con tenencia predeterminada que pueda conectarse a una VPC dedicada.

Para obtener más información, consulte [Instancias dedicadas](#) en la Guía del usuario de EC2 para instancias de Linux y [¿Cómo se conecta una función de Lambda a una VPC dedicada?](#) del Centro de conocimientos de AWS.

Las funciones de Lambda creadas por AWS ParallelCluster se pueden conectar a una VPC privada. Estas funciones de Lambda necesitan acceder a los Servicios de AWS. Puede proporcionarles acceso a través de Internet o puntos de conexión de VPC mediante los siguientes métodos.

- Acceso a Internet

Para acceder a Internet y los Servicios de AWS, una función de Lambda requiere la traducción de direcciones de red (NAT). Enrute el tráfico saliente desde su subred privada a la [puerta de enlace de NAT](#) en una subred pública.

- Puntos de conexión de la VPC

Varios servicios de AWS ofrecen [puntos de enlace de la VPC](#). Puede utilizar los puntos de conexión de VPC para conectarse a Servicios de AWS desde una VPC que no tenga acceso a Internet. Para ver la lista de puntos de conexión de VPC de AWS ParallelCluster, consulte [Redes](#).

Note

Todas las combinaciones de subredes y grupos de seguridad deben permitir el acceso a los Servicios de AWS mediante uno de estos métodos. Las subredes y los grupos de seguridad deben estar en la misma VPC.

Para obtener más información, consulte los [puntos de conexión de VPC](#) en la Guía del usuario de Amazon Virtual Private Cloud y el [Acceso a Internet y a los servicios para las funciones conectadas a la VPC](#) en la Guía para desarrolladores de AWS Lambda.

Para configurar el uso de funciones de Lambda y de las VPC, consulte [DeploymentSettings/LambdaFunctionsVpcConfig](#) para clústeres o [DeploymentSettings/LambdaFunctionsVpcConfig](#) para imágenes.

AWS Identity and Access Management permisos en AWS ParallelCluster

AWS ParallelCluster utiliza los permisos de IAM para controlar el acceso a los recursos al crear y gestionar clústeres.

Para crear y administrar clústeres en una AWS cuenta, se AWS ParallelCluster requieren permisos en dos niveles:

- Permisos que el usuario de `pcluster` necesita a la hora de invocar los comandos CLI de `pcluster` para crear y administrar clústeres.
- Permisos que los recursos del clúster requieren para realizar las acciones del clúster.

AWS ParallelCluster utiliza un [perfil y un rol de instancia EC2](#) para proporcionar permisos de recursos de clúster. Para administrar los permisos de los recursos del clúster, AWS ParallelCluster también requiere permisos para los recursos de IAM. Para obtener más información, consulte [AWS ParallelCluster ejemplos de políticas de usuario para administrar los recursos de IAM](#).

Los usuarios de `pcluster` necesitan permisos de IAM para usar la CLI `pcluster` con el fin de crear y administrar un clúster y sus recursos. Estos permisos se incluyen en las políticas de IAM que se pueden añadir a un usuario o rol. Para obtener más información sobre los roles de IAM,

consulte [Creación de un rol de usuario](#) en la Guía del usuario de IAM de AWS Identity and Access Management .

También puede utilizar [AWS ParallelCluster parámetros de configuración para gestionar los permisos de IAM](#).

Las siguientes secciones contienen los permisos necesarios con ejemplos.

Para usar las políticas de ejemplo sustituya `<REGION>`, `<AWS ACCOUNT ID>` y las cadenas similares por los valores adecuados.

Las siguientes políticas de ejemplo incluyen nombres de recursos de Amazon (ARN) para los recursos. Si trabaja en las particiones AWS GovCloud (US) o en AWS China, debe cambiar los ARN. En concreto, deben cambiarse de «arn:aws» a «arn:aws-us-gov» para la AWS GovCloud (US) partición o «arn:aws-cn» para la partición de China. AWS Para obtener más información, consulte [Nombres de recursos de Amazon \(ARN\) en AWS GovCloud \(US\) regiones](#) en la Guía del AWS GovCloud (US) usuario y [ARN de AWS servicios en China en](#) Introducción a AWS los servicios en China.

Puede realizar un seguimiento de los cambios en las políticas de ejemplo que aparecen en la [AWS ParallelCluster documentación sobre](#). GitHub

Temas

- [AWS ParallelCluster Funciones de instancia EC2](#)
- [AWS ParallelCluster ejemplos pcluster de políticas de usuario](#)
- [AWS ParallelCluster ejemplos de políticas de usuario para administrar los recursos de IAM](#)
- [AWS ParallelCluster parámetros de configuración para gestionar los permisos de IAM](#)

AWS ParallelCluster Funciones de instancia EC2

Al crear un clúster con los valores de configuración predeterminados, AWS ParallelCluster utiliza los [perfiles de instancia](#) de EC2 para crear automáticamente un [rol de instancia](#) de EC2 de clúster predeterminado que proporciona los permisos necesarios para crear y administrar el clúster y sus recursos.

Alternativas al uso del rol de instancia predeterminado AWS ParallelCluster

En lugar del rol de AWS ParallelCluster instancia predeterminado, puede usar la configuración del InstanceRole clúster para especificar su propio rol de IAM existente para EC2. Para obtener más

información, consulte [AWS ParallelCluster parámetros de configuración para gestionar los permisos de IAM](#). Por lo general, debe especificar los roles de IAM existentes para controlar completamente los permisos concedidos a EC2.

Si su intención es añadir políticas adicionales al rol de instancia predeterminado, le recomendamos que apruebe las políticas de IAM adicionales utilizando la configuración de [AdditionalIamPolicies](#) en lugar de la de [InstanceProfile o InstanceRole](#). Cuando actualice el clúster, puede actualizar las `AdditionalIamPolicies`, pero no el `InstanceRole`.

AWS ParallelCluster ejemplos `pcluster` de políticas de usuario

Los siguientes ejemplos muestran las políticas de usuario necesarias para crear AWS ParallelCluster y administrar sus recursos mediante la `pcluster` CLI. Ahora puede asociar políticas a un usuario o rol.

Temas

- [Política de usuario básica `pcluster` de AWS ParallelCluster](#)
- [Política de usuario adicional `pcluster` de AWS ParallelCluster al utilizar el programador AWS Batch](#)
- [Política de usuario adicional `pcluster` de AWS ParallelCluster al utilizar Amazon FSx para Lustre](#)
- [AWS ParallelCluster política de `pcluster` usuario: creación de imágenes](#)

Política de usuario básica `pcluster` de AWS ParallelCluster

La siguiente política muestra los permisos necesarios para ejecutar AWS ParallelCluster `pcluster` comandos.

La última acción de la política se incluye para validar cualquier secreto especificado en la configuración del clúster. Por ejemplo, se utiliza un AWS Secrets Manager secreto para configurar la [DirectoryService](#) integración. En este caso, se crea un clúster solo si existe un secreto válido en [PasswordSecretArn](#). Si se omite esta acción, se omite la validación del secreto. Para mejorar su postura de seguridad, le recomendamos que amplíe el alcance de esta declaración de política añadiendo únicamente los secretos especificados en la configuración del clúster.

Note

Si los sistemas de archivos de Amazon EFS existentes son los únicos sistemas de archivos que se utilizan en su clúster, puede limitar las declaraciones de política de Amazon EFS de

ejemplo a los sistemas de archivos específicos a los que se hace referencia en la [Sección de SharedStorage](#) del archivo de configuración del clúster.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:Describe*"
      ],
      "Resource": "*",
      "Effect": "Allow",
      "Sid": "EC2Read"
    },
    {
      "Action": [
        "ec2:AllocateAddress",
        "ec2:AssociateAddress",
        "ec2:AttachNetworkInterface",
        "ec2:AuthorizeSecurityGroupEgress",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:CreateFleet",
        "ec2:CreateLaunchTemplate",
        "ec2:CreateLaunchTemplateVersion",
        "ec2:CreateNetworkInterface",
        "ec2:CreatePlacementGroup",
        "ec2:CreateSecurityGroup",
        "ec2:CreateSnapshot",
        "ec2:CreateTags",
        "ec2>DeleteTags",
        "ec2:CreateVolume",
        "ec2>DeleteLaunchTemplate",
        "ec2>DeleteNetworkInterface",
        "ec2>DeletePlacementGroup",
        "ec2>DeleteSecurityGroup",
        "ec2>DeleteVolume",
        "ec2:DisassociateAddress",
        "ec2:ModifyLaunchTemplate",
        "ec2:ModifyNetworkInterfaceAttribute",
        "ec2:ModifyVolume",
        "ec2:ModifyVolumeAttribute",
        "ec2:ReleaseAddress",
```



```

        "ec2:RevokeSecurityGroupEgress",
        "ec2:RevokeSecurityGroupIngress",
        "ec2:RunInstances",
        "ec2:TerminateInstances"
    ],
    "Resource": "*",
    "Effect": "Allow",
    "Sid": "EC2Write"
},
{
    "Action": [
        "dynamodb:DescribeTable",
        "dynamodb:ListTagsOfResource",
        "dynamodb:CreateTable",
        "dynamodb>DeleteTable",
        "dynamodb:GetItem",
        "dynamodb:PutItem",
        "dynamodb:UpdateItem",
        "dynamodb:Query",
        "dynamodb:TagResource"
    ],
    "Resource": "arn:aws:dynamodb:*:<AWS ACCOUNT ID>:table/parallelcluster-*",
    "Effect": "Allow",
    "Sid": "DynamoDB"
},
{
    "Action": [
        "route53:ChangeResourceRecordSets",
        "route53:ChangeTagsForResource",
        "route53:CreateHostedZone",
        "route53>DeleteHostedZone",
        "route53:GetChange",
        "route53:GetHostedZone",
        "route53:ListResourceRecordSets",
        "route53:ListQueryLoggingConfigs"
    ],
    "Resource": "*",
    "Effect": "Allow",
    "Sid": "Route53HostedZones"
},
{
    "Action": [
        "cloudformation:*"
    ],

```

```

    "Resource": "*",
    "Effect": "Allow",
    "Sid": "CloudFormation"
  },
  {
    "Action": [
      "cloudwatch:PutDashboard",
      "cloudwatch:ListDashboards",
      "cloudwatch>DeleteDashboards",
      "cloudwatch:GetDashboard",
      "cloudwatch:PutMetricAlarm",
      "cloudwatch>DeleteAlarms",
      "cloudwatch:DescribeAlarms"
      "cloudwatch:PutCompositeAlarm"
    ],
    "Resource": "*",
    "Effect": "Allow",
    "Sid": "CloudWatch"
  },
  {
    "Action": [
      "iam:GetRole",
      "iam:GetRolePolicy",
      "iam:GetPolicy",
      "iam:SimulatePrincipalPolicy",
      "iam:GetInstanceProfile"
    ],
    "Resource": [
      "arn:aws:iam::<AWS ACCOUNT ID>:role/*",
      "arn:aws:iam::<AWS ACCOUNT ID>:policy/*",
      "arn:aws:iam::aws:policy/*",
      "arn:aws:iam::<AWS ACCOUNT ID>:instance-profile/*"
    ],
    "Effect": "Allow",
    "Sid": "IamRead"
  },
  {
    "Action": [
      "iam:CreateInstanceProfile",
      "iam>DeleteInstanceProfile",
      "iam:AddRoleToInstanceProfile",
      "iam:RemoveRoleFromInstanceProfile"
    ],
    "Resource": [

```

```

        "arn:aws:iam::<AWS ACCOUNT ID>:instance-profile/parallelcluster/*"
    ],
    "Effect": "Allow",
    "Sid": "IamInstanceProfile"
},
{
    "Condition": {
        "StringEqualsIfExists": {
            "iam:PassedToService": [
                "lambda.amazonaws.com",
                "ec2.amazonaws.com",
                "spotfleet.amazonaws.com"
            ]
        }
    },
    "Action": [
        "iam:PassRole"
    ],
    "Resource": [
        "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*"
    ],
    "Effect": "Allow",
    "Sid": "IamPassRole"
},
{
    "Action": [
        "lambda:CreateFunction",
        "lambda>DeleteFunction",
        "lambda:GetFunctionConfiguration",
        "lambda:GetFunction",
        "lambda:InvokeFunction",
        "lambda:AddPermission",
        "lambda:RemovePermission",
        "lambda:UpdateFunctionConfiguration",
        "lambda:TagResource",
        "lambda:ListTags",
        "lambda:UntagResource"
    ],
    "Resource": [
        "arn:aws:lambda:*:<AWS ACCOUNT ID>:function:parallelcluster-*",
        "arn:aws:lambda:*:<AWS ACCOUNT ID>:function:pcluster-*"
    ],
    "Effect": "Allow",
    "Sid": "Lambda"
}

```

```
    },
    {
      "Action": [
        "s3:*"
      ],
      "Resource": [
        "arn:aws:s3:::parallelcluster-*",
        "arn:aws:s3:::aws-parallelcluster-*"
      ],
      "Effect": "Allow",
      "Sid": "S3ResourcesBucket"
    },
    {
      "Action": [
        "s3:Get*",
        "s3:List*"
      ],
      "Resource": "arn:aws:s3:::*-aws-parallelcluster*",
      "Effect": "Allow",
      "Sid": "S3ParallelClusterReadOnly"
    },
    {
      "Action": [
        "elasticfilesystem:*"
      ],
      "Resource": [
        "arn:aws:elasticfilesystem:*:<AWS ACCOUNT ID>:*"
      ],
      "Effect": "Allow",
      "Sid": "EFS"
    },
    {
      "Action": [
        "logs:DeleteLogGroup",
        "logs:PutRetentionPolicy",
        "logs:DescribeLogGroups",
        "logs:CreateLogGroup",
        "logs:TagResource",
        "logs:UntagResource",
        "logs:FilterLogEvents",
        "logs:GetLogEvents",
        "logs:CreateExportTask",
        "logs:DescribeLogStreams",
        "logs:DescribeExportTasks",
```

```

        "logs:DescribeMetricFilters",
        "logs:PutMetricFilter",
        "logs>DeleteMetricFilter"
    ],
    "Resource": "*",
    "Effect": "Allow",
    "Sid": "CloudWatchLogs"
},
{
    "Action": [
        "resource-groups:ListGroupResources"
    ],
    "Resource": "*",
    "Effect": "Allow",
    "Sid": "ResourceGroupRead"
},
{
    "Sid": "AllowDescribingFileCache",
    "Effect": "Allow",
    "Action": [
        "fsx:DescribeFileCaches"
    ],
    "Resource": "*"
},
{
    "Action": "secretsmanager:DescribeSecret",
    "Resource": "arn:aws:secretsmanager:<REGION>:<AWS ACCOUNT ID>:secret:<SECRET
NAME>",
    "Effect": "Allow"
}
]
}

```

Política de usuario adicional **pcluster** de AWS ParallelCluster al utilizar el programador AWS Batch

En caso de que necesite crear y administrar un clúster con un AWS Batch programador, se requiere la siguiente política adicional.

```

{
    "Version": "2012-10-17",
    "Statement": [
        {

```

```

    "Condition": {
      "StringEqualsIfExists": {
        "iam:PassedToService": [
          "ecs-tasks.amazonaws.com",
          "batch.amazonaws.com",
          "codebuild.amazonaws.com"
        ]
      }
    },
    "Action": [
      "iam:PassRole"
    ],
    "Resource": [
      "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*"
    ],
    "Effect": "Allow",
    "Sid": "IamPassRole"
  },
  {
    "Condition": {
      "StringEquals": {
        "iam:AWSServiceName": [
          "batch.amazonaws.com"
        ]
      }
    },
    "Action": [
      "iam:CreateServiceLinkedRole",
      "iam>DeleteServiceLinkedRole"
    ],
    "Resource": [
      "arn:aws:iam::<AWS ACCOUNT ID>:role/aws-service-role/
batch.amazonaws.com/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "codebuild:*"
    ],
    "Resource": "arn:aws:codebuild:*:<AWS ACCOUNT ID>:project/pcluster-*",
    "Effect": "Allow"
  },
  {

```

```
    "Action": [
      "ecr:*"
    ],
    "Resource": "*",
    "Effect": "Allow",
    "Sid": "ECR"
  },
  {
    "Action": [
      "batch:*"
    ],
    "Resource": "*",
    "Effect": "Allow",
    "Sid": "Batch"
  },
  {
    "Action": [
      "events:*"
    ],
    "Resource": "*",
    "Effect": "Allow",
    "Sid": "AmazonCloudWatchEvents"
  },
  {
    "Action": [
      "ecs:DescribeContainerInstances",
      "ecs:ListContainerInstances"
    ],
    "Resource": "*",
    "Effect": "Allow",
    "Sid": "ECS"
  }
]
```

Política de usuario adicional **pcluster** de AWS ParallelCluster al utilizar Amazon FSx para Lustre

En caso de que necesite crear y administrar un clúster con Amazon FSx para Lustre, se requiere la siguiente política adicional.

Note

Si los sistemas de archivos de Amazon FSx existentes son los únicos sistemas de archivos que se utilizan en su clúster, puede limitar las declaraciones de política de Amazon FSx de ejemplo a los sistemas de archivos específicos a los que se hace referencia en la [Sección de SharedStorage](#) del archivo de configuración del clúster.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Condition": {
        "StringEquals": {
          "iam:AWSServiceName": [
            "fsx.amazonaws.com",
            "s3.data-source.lustre.fsx.amazonaws.com"
          ]
        }
      },
      "Action": [
        "iam:CreateServiceLinkedRole",
        "iam>DeleteServiceLinkedRole"
      ],
      "Resource": "*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "fsx:*"
      ],
      "Resource": [
        "arn:aws:fsx:*:<AWS ACCOUNT ID>:*"
      ],
      "Effect": "Allow",
      "Sid": "FSx"
    },
    {
      "Action": [
        "iam:CreateServiceLinkedRole",
        "iam:AttachRolePolicy",
        "iam:PutRolePolicy"
      ]
    }
  ]
}
```



```

    ],
    "Resource": "arn:aws:iam::<AWS ACCOUNT ID>:role/aws-service-role/s3.data-
source.lustre.fsx.amazonaws.com/*",
    "Effect": "Allow"
  },
  {
    "Action": [
      "s3:Get*",
      "s3:List*",
      "s3:PutObject"
    ],
    "Resource": "arn:aws:s3:::<S3 NAME>",
    "Effect": "Allow"
  }
]
}

```

AWS ParallelCluster política de **pcluster** usuario: creación de imágenes

Los usuarios que pretendan crear imágenes EC2 personalizadas AWS ParallelCluster deben tener el siguiente conjunto de permisos.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:DescribeImages",
        "ec2:DescribeInstanceTypeOfferings",
        "ec2:DescribeInstanceTypes",
        "ec2:DeregisterImage",
        "ec2>DeleteSnapshot"
      ],
      "Resource": "*",
      "Effect": "Allow",
      "Sid": "EC2"
    },
    {
      "Action": [
        "iam:CreateInstanceProfile",
        "iam:AddRoleToInstanceProfile",
        "iam:GetRole",
        "iam:GetRolePolicy",

```

```

        "iam:GetInstanceProfile",
        "iam:RemoveRoleFromInstanceProfile"
    ],
    "Resource": [
        "arn:aws:iam::<AWS ACCOUNT ID>:instance-profile/parallelcluster/*",
        "arn:aws:iam::<AWS ACCOUNT ID>:instance-profile/ParallelClusterImage*",
        "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*"
    ],
    "Effect": "Allow",
    "Sid": "IAM"
},
{
    "Condition": {
        "StringEquals": {
            "iam:PassedToService": [
                "lambda.amazonaws.com",
                "ec2.amazonaws.com"
            ]
        }
    },
    "Action": [
        "iam:PassRole"
    ],
    "Resource": [
        "arn:aws:iam::<AWS ACCOUNT ID>:instance-profile/parallelcluster/*",
        "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*"
    ],
    "Effect": "Allow",
    "Sid": "IAMPassRole"
},
{
    "Action": [
        "logs:CreateLogGroup",
        "logs:TagResource",
        "logs:UntagResource",
        "logs>DeleteLogGroup"
    ],
    "Resource": [
        "arn:aws:logs:*:<AWS ACCOUNT ID>:log-group:/aws/imagebuilder/ParallelClusterImage-*",
        "arn:aws:logs:*:<AWS ACCOUNT ID>:log-group:/aws/lambda/ParallelClusterImage-*"
    ],
    "Effect": "Allow",

```

```
    "Sid": "CloudWatch"
  },
  {
    "Action": [
      "cloudformation:DescribeStacks",
      "cloudformation:CreateStack",
      "cloudformation>DeleteStack"
    ],
    "Resource": [
      "arn:aws:cloudformation:*:<AWS ACCOUNT ID>:stack/*"
    ],
    "Effect": "Allow",
    "Sid": "CloudFormation"
  },
  {
    "Action": [
      "lambda:CreateFunction",
      "lambda:GetFunction",
      "lambda:AddPermission",
      "lambda:RemovePermission",
      "lambda>DeleteFunction",
      "lambda:TagResource",
      "lambda:ListTags",
      "lambda:UntagResource"
    ],
    "Resource": [
      "arn:aws:lambda:*:<AWS ACCOUNT ID>:function:ParallelClusterImage-*"
    ],
    "Effect": "Allow",
    "Sid": "Lambda"
  },
  {
    "Action": [
      "imagebuilder:Get*"
    ],
    "Resource": "*",
    "Effect": "Allow",
    "Sid": "ImageBuilderGet"
  },
  {
    "Action": [
      "imagebuilder:CreateImage",
      "imagebuilder:TagResource",
      "imagebuilder:CreateImageRecipe",
```

```

        "imagebuilder:CreateComponent",
        "imagebuilder:CreateDistributionConfiguration",
        "imagebuilder:CreateInfrastructureConfiguration",
        "imagebuilder>DeleteImage",
        "imagebuilder>DeleteComponent",
        "imagebuilder>DeleteImageRecipe",
        "imagebuilder>DeleteInfrastructureConfiguration",
        "imagebuilder>DeleteDistributionConfiguration"
    ],
    "Resource": [
        "arn:aws:imagebuilder:*:<AWS ACCOUNT ID>:image/parallelclusterimage-*",
        "arn:aws:imagebuilder:*:<AWS ACCOUNT ID>:image-recipe/parallelclusterimage-*",
        "arn:aws:imagebuilder:*:<AWS ACCOUNT ID>:component/parallelclusterimage-*",
        "arn:aws:imagebuilder:*:<AWS ACCOUNT ID>:distribution-configuration/parallelclusterimage-*",
        "arn:aws:imagebuilder:*:<AWS ACCOUNT ID>:infrastructure-configuration/parallelclusterimage-*"
    ],
    "Effect": "Allow",
    "Sid": "ImageBuilder"
},
{
    "Action": [
        "s3:CreateBucket",
        "s3:ListBucket",
        "s3:ListBucketVersions"
    ],
    "Resource": [
        "arn:aws:s3:::parallelcluster-*"
    ],
    "Effect": "Allow",
    "Sid": "S3Bucket"
},
{
    "Action": [
        "sns:GetTopicAttributes",
        "sns:TagResource",
        "sns:CreateTopic",
        "sns:Subscribe",
        "sns:Publish",
        "SNS>DeleteTopic",
        "SNS:Unsubscribe"
    ]
}

```

```

    ],
    "Resource": [
      "arn:aws:sns:*:<AWS ACCOUNT ID>:ParallelClusterImage-*"
    ],
    "Effect": "Allow",
    "Sid": "SNS"
  },
  {
    "Action": [
      "s3:PutObject",
      "s3:GetObject",
      "s3:GetObjectVersion",
      "s3:DeleteObject",
      "s3:DeleteObjectVersion"
    ],
    "Resource": [
      "arn:aws:s3:::parallelcluster-*/*"
    ],
    "Effect": "Allow",
    "Sid": "S3Objects"
  },
  {
    "Action": "iam:CreateServiceLinkedRole",
    "Effect": "Allow",
    "Resource": "arn:aws:iam::*:role/aws-service-role/
imagebuilder.amazonaws.com/AWSServiceRoleForImageBuilder",
    "Condition": {
      "StringLike": {
        "iam:AWSServiceName": "imagebuilder.amazonaws.com"
      }
    }
  }
]
}

```

AWS ParallelCluster ejemplos de políticas de usuario para administrar los recursos de IAM

Cuando se utilizan AWS ParallelCluster para crear clústeres o AMI personalizadas, se deben proporcionar políticas de IAM que contengan permisos para conceder el conjunto de permisos necesario a AWS ParallelCluster los componentes. Estos recursos de IAM pueden crearse

automáticamente AWS ParallelCluster o proporcionarse como entrada al crear un clúster o una imagen personalizada.

Puede utilizar los siguientes modos para proporcionar al AWS ParallelCluster usuario los permisos necesarios para acceder a los recursos de IAM mediante el uso de políticas de IAM adicionales en la configuración.

Temas

- [Modo de acceso de IAM con privilegios](#)
- [Modo de acceso de IAM restringido](#)
- [Modo PermissionsBoundary](#)

Modo de acceso de IAM con privilegios

Con este modo, crea AWS ParallelCluster automáticamente todos los recursos de IAM necesarios. Estas políticas de IAM tienen un alcance limitado para permitir el acceso únicamente a los recursos del clúster.

Para habilitar el modo de acceso de IAM con privilegios, añada la siguiente política al rol de usuario.

Note

Si configura [AdditionalPolicies](#) los parámetros [HeadNodeIam/AdditionalPolicies](#) o [Scheduling//SlurmQueuesIam/](#), debe proporcionar al AWS ParallelCluster usuario permiso para adjuntar y desvincular políticas de rol para cada política adicional, tal y como se muestra en la siguiente política. Agregue los ARN de política adicionales a la condición para asociar y desvincular políticas de rol.

Warning

Este modo permite al usuario tener privilegios de administrador de IAM en Cuenta de AWS

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```

    "Action": [
      "iam:CreateServiceLinkedRole",
      "iam>DeleteRole",
      "iam:TagRole"
    ],
    "Resource": [
      "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*"
    ],
    "Effect": "Allow",
    "Sid": "IamRole"
  },
  {
    "Action": [
      "iam>CreateRole"
    ],
    "Resource": [
      "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*"
    ],
    "Effect": "Allow",
    "Sid": "IamCreateRole"
  },
  {
    "Action": [
      "iam:PutRolePolicy",
      "iam>DeleteRolePolicy"
    ],
    "Resource": "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*",
    "Effect": "Allow",
    "Sid": "IamInlinePolicy"
  },
  {
    "Condition": {
      "ArnLike": {
        "iam:PolicyARN": [
          "arn:aws:iam::<AWS ACCOUNT ID>:policy/parallelcluster*",
          "arn:aws:iam::<AWS ACCOUNT ID>:policy/parallelcluster/*",
          "arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy",
          "arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore",
          "arn:aws:iam::aws:policy/AWSBatchFullAccess",
          "arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess",
          "arn:aws:iam::aws:policy/service-role/AWSBatchServiceRole",
          "arn:aws:iam::aws:policy/service-role/
AmazonEC2ContainerServiceforEC2Role",

```

```

        "arn:aws:iam::aws:policy/service-role/
AmazonECSTaskExecutionRolePolicy",
        "arn:aws:iam::aws:policy/service-role/
AmazonEC2SpotFleetTaggingRole",
        "arn:aws:iam::aws:policy/EC2InstanceProfileForImageBuilder",
        "arn:aws:iam::aws:policy/service-role/
AWSLambdaBasicExecutionRole"
    ]
  }
},
"Action": [
  "iam:AttachRolePolicy",
  "iam:DetachRolePolicy"
],
"Resource": "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*",
"Effect": "Allow",
"Sid": "IamPolicy"
}
]
}

```

Modo de acceso de IAM restringido

Cuando no se otorgan políticas de IAM adicionales al usuario, un administrador debe crear manualmente los roles de IAM requeridos por los clústeres o la creación de imágenes personalizadas y pasarlas como parte de la configuración del clúster.

Al crear un clúster, se requieren los siguientes parámetros:

- [Iam / Roles / LambdaFunctionsRole](#)
- [HeadNode / Iam / InstanceRole](#) | [InstanceProfile](#)
- [Scheduling / SlurmQueues / Iam / InstanceRole](#) | [InstanceProfile](#)

Al crear una imagen personalizada, se requieren los siguientes parámetros:

- [Build / Iam / InstanceRole](#) | [InstanceProfile](#)
- [Build / Iam / CleanupLambdaRole](#)

Los roles de IAM transferidos como parte de los parámetros mencionados anteriormente deben crearse con el prefijo de ruta `/parallelcluster/`. Si esto no es posible, es necesario actualizar la

política de usuario para conceder permiso de `iam:PassRole` a los roles personalizados específicos, como en el siguiente ejemplo.

```
{
  "Condition": {
    "StringEqualsIfExists": {
      "iam:PassedToService": [
        "ecs-tasks.amazonaws.com",
        "lambda.amazonaws.com",
        "ec2.amazonaws.com",
        "spotfleet.amazonaws.com",
        "batch.amazonaws.com",
        "codebuild.amazonaws.com"
      ]
    }
  },
  "Action": [
    "iam:PassRole"
  ],
  "Resource": [
    <list all custom IAM roles>
  ],
  "Effect": "Allow",
  "Sid": "IamPassRole"
}
```

Warning

Actualmente, este modo no permite la administración de AWS Batch clústeres porque no todas las funciones de IAM se pueden transferir a la configuración del clúster.

Modo **PermissionsBoundary**

Este modo se delega AWS ParallelCluster en la creación de funciones de IAM vinculadas al límite de permisos de IAM configurado. Para obtener más información sobre los límites de los permisos de IAM, consulte [Límites de permisos para las entidades de IAM](#) en la Guía del usuario de IAM.

Se debe agregar la siguiente política al rol de usuario.

En la política, sustituya `< permissions-boundary-arn >` por el ARN de la política de IAM que se aplicará como límite de permisos.

⚠ Warning

Si configura los parámetros [HeadNode/Iam/AdditionalPolicies](#) o [Scheduling/SlurmQueues/Iam/AdditionalPolicies](#), debe conceder al usuario permiso para asociar y desvincular políticas de rol para cada política adicional, tal y como se muestra en la siguiente política. Agregue los ARN de política adicionales a la condición para asociar y desvincular políticas de rol.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "iam:CreateServiceLinkedRole",
        "iam:DeleteRole",
        "iam:TagRole"
      ],
      "Resource": [
        "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*"
      ],
      "Effect": "Allow",
      "Sid": "IamRole"
    },
    {
      "Condition": {
        "StringEquals": {
          "iam:PermissionsBoundary": [
            <permissions-boundary-arn>
          ]
        }
      },
      "Action": [
        "iam:CreateRole"
      ],
      "Resource": [
        "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*"
      ],
      "Effect": "Allow",
      "Sid": "IamCreateRole"
    }
  ],
  {
```

```

    "Condition": {
      "StringEquals": {
        "iam:PermissionsBoundary": [
          <permissions-boundary-arn>
        ]
      }
    },
    "Action": [
      "iam:PutRolePolicy",
      "iam>DeleteRolePolicy"
    ],
    "Resource": "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*",
    "Effect": "Allow",
    "Sid": "IamInlinePolicy"
  },
  {
    "Condition": {
      "StringEquals": {
        "iam:PermissionsBoundary": [
          <permissions-boundary-arn>
        ]
      },
      "ArnLike": {
        "iam:PolicyARN": [
          "arn:aws:iam::<AWS ACCOUNT ID>:policy/parallelcluster*",
          "arn:aws:iam::<AWS ACCOUNT ID>:policy/parallelcluster/*",
          "arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy",
          "arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore",
          "arn:aws:iam::aws:policy/AWSBatchFullAccess",
          "arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess",
          "arn:aws:iam::aws:policy/service-role/AWSBatchServiceRole",
          "arn:aws:iam::aws:policy/service-role/
AmazonEC2ContainerServiceforEC2Role",
          "arn:aws:iam::aws:policy/service-role/
AmazonECSTaskExecutionRolePolicy",
          "arn:aws:iam::aws:policy/service-role/
AmazonEC2SpotFleetTaggingRole",
          "arn:aws:iam::aws:policy/EC2InstanceProfileForImageBuilder",
          "arn:aws:iam::aws:policy/service-role/
AWSLambdaBasicExecutionRole"
        ]
      }
    },
    "Action": [

```

```
        "iam:AttachRolePolicy",
        "iam:DetachRolePolicy"
    ],
    "Resource": "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*",
    "Effect": "Allow",
    "Sid": "IamPolicy"
}
]
```

Cuando este modo está habilitado, debe especificar el ARN del límite de permisos en el parámetro de configuración [Iam/PermissionsBoundary](#) al crear o actualizar un clúster y en el parámetro [Build/Iam/PermissionBoundary](#) al crear una imagen personalizada.

AWS ParallelCluster parámetros de configuración para gestionar los permisos de IAM

AWS ParallelCluster expone una serie de opciones de configuración para personalizar y administrar los permisos y funciones de IAM que se utilizan en un clúster o durante el proceso de creación de una AMI personalizada.

Temas

- [Configuración del clúster](#)
- [Configuración de imagen personalizada](#)

Configuración del clúster

Temas

- [Rol de IAM del nodo principal](#)
- [Acceso a Amazon S3](#)
- [Políticas de IAM adicionales](#)
- [AWS Lambda función: rol](#)
- [Rol de IAM de nodos de computación](#)
- [Límite de permisos](#)

Rol de IAM del nodo principal

[HeadNode](#) / [Iam](#) / [InstanceRole](#) | [InstanceProfile](#)

Con esta opción, se anula el rol de IAM predeterminado que se asigna al nodo principal del clúster. Para obtener más información, consulte la referencia del [InstanceProfile](#).

Este es el conjunto mínimo de políticas que se utilizarán como parte de este rol cuando el programador sea Slurm:

- Política de IAM administrada por `arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy`. Para obtener más información, consulte [Crear roles y usuarios de IAM para usarlos con el CloudWatch agente](#) en la Guía del CloudWatch usuario de Amazon.
- Política de IAM administrada por `arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore`. Para obtener más información, consulte [Políticas administradas por AWS para AWS Systems Manager](#) en la Guía del usuario de AWS Systems Manager .
- Política de IAM adicional:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion"
      ],
      "Resource": [
        "arn:aws:s3:::<REGION>-aws-parallelcluster/*",
        "arn:aws:s3:::dcv-license.<REGION>/*",
        "arn:aws:s3:::parallelcluster-*v1-do-not-delete/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "dynamodb:GetItem",
        "dynamodb:PutItem",
        "dynamodb:UpdateItem",
        "dynamodb:BatchWriteItem",
        "dynamodb:BatchGetItem"
      ]
    }
  ]
}
```

```

    ],
    "Resource": "arn:aws:dynamodb:<REGION>:<AWS ACCOUNT ID>:table/
parallelcluster-*",
    "Effect": "Allow"
  },
  {
    "Condition": {
      "StringEquals": {
        "ec2:ResourceTag/parallelcluster:node-type": "Compute"
      }
    },
    "Action": "ec2:TerminateInstances",
    "Resource": "*",
    "Effect": "Allow"
  },
  {
    "Action": [
      "ec2:RunInstances",
      "ec2:CreateFleet"
    ],
    "Resource": "*",
    "Effect": "Allow"
  },
  {
    "Condition": {
      "StringEquals": {
        "iam:PassedToService": [
          "ec2.amazonaws.com"
        ]
      }
    },
    "Action": [
      "iam:PassRole"
    ],
    "Resource": [
      "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*",
      "arn:aws:iam::<AWS ACCOUNT ID>:instance-profile/parallelcluster/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "ec2:DescribeInstances",
      "ec2:DescribeInstanceStatus",

```

```

        "ec2:DescribeVolumes",
        "ec2:DescribeInstanceAttribute",
        "ec2:DescribeCapacityReservations"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Action": [
        "ec2:CreateTags",
        "ec2:AttachVolume"
    ],
    "Resource": [
        "arn:aws:ec2:<REGION>:<AWS ACCOUNT ID>:instance/*",
        "arn:aws:ec2:<REGION>:<AWS ACCOUNT ID>:volume/*"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "cloudformation:DescribeStacks",
        "cloudformation:DescribeStackResource",
        "cloudformation:SignalResource"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Action": [
        "route53:ChangeResourceRecordSets"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Action": "secretsmanager:GetSecretValue",
    "Resource": "arn:aws:secretsmanager:<REGION>:<AWS ACCOUNT ID>:secret:<SECRET_ID>",
    "Effect": "Allow"
}
]
}

```

Tenga en cuenta que, en caso de que se utilice [Scheduling/SlurmQueues/Iam/InstanceRole](#) para anular el rol de IAM de computación, la política del nodo principal descrita anteriormente debe incluir dicho rol en la sección Resource del permiso `iam:PassRole`.

Este es el conjunto mínimo de políticas que se utilizarán como parte de este rol cuando el programador sea AWS Batch:

- Política de IAM administrada por `arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy`. Para obtener más información, consulte [Crear roles y usuarios de IAM para usarlos con el CloudWatch agente](#) en la Guía del CloudWatch usuario de Amazon.
- Política de IAM administrada por `arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore`. Para obtener más información, consulte [Políticas administradas por AWS para AWS Systems Manager](#) en la Guía del usuario de AWS Systems Manager .
- Política de IAM adicional:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:GetObjectVersion"
      ],
      "Resource": [
        "arn:aws:s3:::parallelcluster-*-*v1-do-not-delete/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": "s3:GetObject",
      "Resource": [
        "arn:aws:s3:::dcv-license.<REGION>/*",
        "arn:aws:s3:::<REGION>-aws-parallelcluster/*"
      ],
      "Effect": "Allow"
    },
    {
      "Condition": {
```



```

        "StringEquals": {
            "iam:PassedToService": [
                "batch.amazonaws.com"
            ]
        }
    },
    "Action": [
        "iam:PassRole"
    ],
    "Resource": [
        "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*",
        "arn:aws:iam::<AWS ACCOUNT ID>:instance-profile/parallelcluster/*"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "batch:DescribeJobQueues",
        "batch:DescribeJobs",
        "batch:ListJobs",
        "batch:DescribeComputeEnvironments"
    ],
    "Resource": "*",
    "Effect": "Allow"
},
{
    "Action": [
        "batch:SubmitJob",
        "batch:TerminateJob",
        "logs:GetLogEvents",
        "ecs:ListContainerInstances",
        "ecs:DescribeContainerInstances",
    ],
    "Resource": [
        "arn:aws:logs:<REGION>:<AWS ACCOUNT ID>:log-group:/aws/batch/job:log-
stream:PclusterJobDefinition*",
        "arn:aws:ecs:<REGION>:<AWS ACCOUNT ID>:container-instance/AWSBatch-
PclusterComputeEnviron*",
        "arn:aws:ecs:<REGION>:<AWS ACCOUNT ID>:cluster/AWSBatch-Pcluster*",
        "arn:aws:batch:<REGION>:<AWS ACCOUNT ID>:job-queue/
PclusterJobQueue*",
        "arn:aws:batch:<REGION>:<AWS ACCOUNT ID>:job-definition/
PclusterJobDefinition*:*",
        "arn:aws:batch:<REGION>:<AWS ACCOUNT ID>:job/*"
    ]
}

```

```

    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "ec2:DescribeInstances",
      "ec2:DescribeInstanceStatus",
      "ec2:DescribeVolumes",
      "ec2:DescribeInstanceAttribute"
    ],
    "Resource": "*",
    "Effect": "Allow"
  },
  {
    "Action": [
      "ec2:CreateTags",
      "ec2:AttachVolume"
    ],
    "Resource": [
      "arn:aws:ec2:<REGION>:<AWS ACCOUNT ID>:instance/*",
      "arn:aws:ec2:<REGION>:<AWS ACCOUNT ID>:volume/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "cloudformation:DescribeStackResource",
      "cloudformation:DescribeStacks",
      "cloudformation:SignalResource"
    ],
    "Resource": "*",
    "Effect": "Allow"
  },
  {
    "Action": "secretsmanager:GetSecretValue",
    "Resource": "arn:aws:secretsmanager:<REGION>:<AWS ACCOUNT ID>:secret:<SECRET_ID>",
    "Effect": "Allow"
  }
]
}

```

Acceso a Amazon S3

[HeadNode/Iam/S3Access](#) o [Scheduling/SlurmQueues/S3Access](#)

En estas secciones de configuración, puede personalizar el acceso a Amazon S3 mediante la concesión de políticas de Amazon S3 adicionales a los roles de IAM asociados al nodo principal o a los nodos de computación del clúster cuando dichos roles los cree AWS ParallelCluster. Consulte la documentación de referencia de cada parámetro de configuración para obtener más información.

Este parámetro solo se puede usar cuando el usuario está configurado con [Modo de acceso de IAM con privilegios](#) o [Modo PermissionsBoundary](#).

Políticas de IAM adicionales

[HeadNode/Iam/AdditionalIamPolicies](#) o [SlurmQueues/Iam/AdditionalIamPolicies](#)

Utilice esta opción para adjuntar políticas de IAM administradas adicionales a las funciones de IAM asociadas al nodo principal o a los nodos de cómputo del clúster cuando dichas funciones las cree. AWS ParallelCluster

Warning

Para utilizar esta opción, asegúrese de que se concedan al [usuario de AWS ParallelCluster](#) los permisos `iam:AttachRolePolicy` y `iam:DetachRolePolicy` necesarios para las políticas de IAM que se deben asociar.

AWS Lambda función: rol

[Iam / Roles / LambdaFunctionsRole](#)

Esta opción anula la función asociada a todas AWS Lambda las funciones que se utilizan durante el proceso de creación del clúster. AWS Lambda debe configurarse como el principal autorizado a asumir el rol.

Note

Si se establece [DeploymentSettings/LambdaFunctionsVpcConfig/](#), `LambdaFunctionsRole` debe incluir el [permiso de rol de AWS Lambda](#) para establecer la configuración de la VPC.

Este es el conjunto mínimo de políticas que se utilizarán como parte de este rol:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "route53:ListResourceRecordSets",
        "route53:ChangeResourceRecordSets"
      ],
      "Resource": "arn:aws:route53::hostedzone/*",
      "Effect": "Allow"
    },
    {
      "Action": ["logs:CreateLogStream", "logs:PutLogEvents"],
      "Effect": "Allow",
      "Resource": "arn:aws:logs:<REGION>:<AWS ACCOUNT ID>:log-group:/aws/lambda/
pcluster-*"
    },
    {
      "Action": "ec2:DescribeInstances",
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Action": "ec2:TerminateInstances",
      "Condition": {
        "StringEquals": {
          "ec2:ResourceTag/parallelcluster:node-type": "Compute"
        }
      },
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Action": [
        "s3:DeleteObject",
        "s3:DeleteObjectVersion",
        "s3:ListBucket",
        "s3:ListBucketVersions"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3::parallelcluster-* -v1-do-not-delete",

```

```

    "arn:aws:s3:::parallelcluster-* -v1-do-not-delete/*"
  ]
}
]
}

```

Rol de IAM de nodos de computación

[Scheduling](#) / [SlurmQueues](#) / [Iam](#) / [InstanceRole](#) | [InstanceProfile](#)

Esta opción permite anular el rol de IAM que se asigna a los nodos de computación del clúster. Para obtener más información, consulte [InstanceProfile](#).

Este es el conjunto mínimo de políticas que se utilizarán como parte de este rol:

- Política de IAM administrada por `arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy`. Para obtener más información, consulte [Crear roles y usuarios de IAM para usarlos con el CloudWatch agente](#) en la Guía del CloudWatch usuario de Amazon.
- Política de IAM administrada por `arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore`. Para obtener más información, consulte [Políticas administradas por AWS para AWS Systems Manager](#) en la Guía del usuario de AWS Systems Manager .
- Política de IAM adicional:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "dynamodb:Query",
        "dynamodb:UpdateItem",
        "dynamodb:PutItem",
        "dynamodb:GetItem"
      ],
      "Resource": "arn:aws:dynamodb:<REGION>:<AWS ACCOUNT ID>:table/parallelcluster-*",
      "Effect": "Allow"
    },
    {
      "Action": "s3:GetObject",

```

```

    "Resource": [
      "arn:aws:s3:::<REGION>-aws-parallelcluster/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": "ec2:DescribeInstanceAttribute",
    "Resource": "*",
    "Effect": "Allow"
  },
  {
    "Action": "cloudformation:DescribeStackResource",
    "Resource":
      [ "arn:aws:cloudformation:<REGION>:<AWS ACCOUNT ID>:stack/*/*" ],
    "Effect": "Allow"
  }
]
}

```

Límite de permisos

[Iam / PermissionsBoundary](#)

Este parámetro obliga AWS ParallelCluster a adjuntar la política de IAM en cuestión como `PermissionsBoundary` a todas las funciones de IAM que se crean como parte de la implementación de un clúster.

Consulte [Modo PermissionsBoundary](#) para conocer la lista de políticas que requiere el usuario cuando se define esta configuración.

Configuración de imagen personalizada

Temas

- [Rol de instancia para Generador de imágenes de EC2](#)
- [AWS Lambda función de limpieza](#)
- [Políticas de IAM adicionales](#)
- [Límite de permisos](#)

Rol de instancia para Generador de imágenes de EC2

[Build](#) / [Iam](#) / [InstanceRole](#) | [InstanceProfile](#)

Con esta opción, anula el rol de IAM asignado a la instancia de EC2 lanzada por Generador de imágenes de EC2 para crear una AMI personalizada.

Este es el conjunto mínimo de políticas que se utilizarán como parte de este rol:

- Política de IAM administrada por `arn:aws:iam::aws:policy/AmazonSSMManagedInstanceCore`. Para obtener más información, consulte [Políticas administradas por AWS para AWS Systems Manager](#) en la Guía del usuario de AWS Systems Manager .
- Política de IAM administrada por `arn:aws:iam::aws:policy/EC2InstanceProfileForImageBuilder`. Para obtener más información, consulte la [política de EC2InstanceProfileForImageBuilder](#) en la Guía del usuario de Generador de imágenes.
- Política de IAM adicional:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ec2:CreateTags",
        "ec2:ModifyImageAttribute"
      ],
      "Resource": "arn:aws:ec2:<REGION>::image/*",
      "Effect": "Allow"
    }
  ]
}
```

AWS Lambda función de limpieza

[Build](#) / [Iam](#) / [CleanupLambdaRole](#)

Esta opción anula la función asociada a todas AWS Lambda las funciones que se utilizan durante el proceso de creación de imágenes personalizadas. AWS Lambda debe configurarse como el principal autorizado a asumir el rol.

Note

Si se establece [DeploymentSettings/LambdaFunctionsVpcConfig/](#), CleanupLambdaRole debe incluir el [permiso de rol de AWS Lambda](#) para establecer la configuración de la VPC.

Este es el conjunto mínimo de políticas que se utilizarán como parte de este rol:

- Política de IAM administrada por `arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole`. Para obtener más información, consulte [las políticas administradas por AWS para las características de Lambda](#) en la Guía para desarrolladores de AWS Lambda .
- Política de IAM adicional:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "iam:DetachRolePolicy",
        "iam>DeleteRole",
        "iam>DeleteRolePolicy"
      ],
      "Resource": "arn:aws:iam::<AWS ACCOUNT ID>:role/parallelcluster/*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "iam>DeleteInstanceProfile",
        "iam:RemoveRoleFromInstanceProfile"
      ],
      "Resource": "arn:aws:iam::<AWS ACCOUNT ID>:instance-profile/parallelcluster/*",
      "Effect": "Allow"
    },
    {
      "Action": "imagebuilder>DeleteInfrastructureConfiguration",
      "Resource": "arn:aws:imagebuilder:<REGION>:<AWS ACCOUNT ID>:infrastructure-configuration/parallelclusterimage-*",
      "Effect": "Allow"
    }
  ]
}
```



```

    },
    {
      "Action": [
        "imagebuilder:DeleteComponent"
      ],
      "Resource": [
        "arn:aws:imagebuilder:<REGION>:<AWS ACCOUNT ID>:component/
parallelclusterimage-*/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": "imagebuilder:DeleteImageRecipe",
      "Resource": "arn:aws:imagebuilder:<REGION>:<AWS ACCOUNT ID>:image-recipe/
parallelclusterimage-*/*",
      "Effect": "Allow"
    },
    {
      "Action": "imagebuilder:DeleteDistributionConfiguration",
      "Resource": "arn:aws:imagebuilder:<REGION>:<AWS ACCOUNT ID>:distribution-
configuration/parallelclusterimage-*",
      "Effect": "Allow"
    },
    {
      "Action": [
        "imagebuilder:DeleteImage",
        "imagebuilder:GetImage",
        "imagebuilder:CancelImageCreation"
      ],
      "Resource": "arn:aws:imagebuilder:<REGION>:<AWS ACCOUNT ID>:image/
parallelclusterimage-*/*",
      "Effect": "Allow"
    },
    {
      "Action": "cloudformation:DeleteStack",
      "Resource": "arn:aws:cloudformation:<REGION>:<AWS ACCOUNT ID>:stack/*/*",
      "Effect": "Allow"
    },
    {
      "Action": "ec2:CreateTags",
      "Resource": "arn:aws:ec2:<REGION>::image/*",
      "Effect": "Allow"
    },
    {

```

```

        "Action": "tag:TagResources",
        "Resource": "*",
        "Effect": "Allow"
    },
    {
        "Action": [
            "lambda:DeleteFunction",
            "lambda:RemovePermission"
        ],
        "Resource": "arn:aws:lambda:<REGION>:<AWS ACCOUNT
ID>:function:ParallelClusterImage-*",
        "Effect": "Allow"
    },
    {
        "Action": "logs:DeleteLogGroup",
        "Resource": "arn:aws:logs:<REGION>:<AWS ACCOUNT ID>:log-group:/aws/
lambda/ParallelClusterImage-*:*",
        "Effect": "Allow"
    },
    {
        "Action": [
            "SNS:GetTopicAttributes",
            "SNS:DeleteTopic",
            "SNS:GetSubscriptionAttributes",
            "SNS:Unsubscribe"
        ],
        "Resource": "arn:aws:sns:<REGION>:<AWS ACCOUNT ID>:ParallelClusterImage-
*",
        "Effect": "Allow"
    }
]
}

```

Políticas de IAM adicionales

[Build](#) / [Iam](#) / [AdditionalIamPolicies](#)

Utilice esta opción para asociar políticas de IAM administradas adicionales al rol relacionado con la instancia de EC2 utilizada por Generador de imágenes de EC2 para generar la AMI personalizada.

⚠ Warning

Para utilizar esta opción, asegúrese de que se concedan al [usuario de AWS ParallelCluster](#) los permisos `iam:AttachRolePolicy` y `iam:DetachRolePolicy` necesarios para las políticas de IAM que se deben asociar.

Límite de permisos

[Build](#) / [Iam](#) / [PermissionsBoundary](#)

Este parámetro obliga AWS ParallelCluster a adjuntar la política de IAM dada `PermissionsBoundary` a todas las funciones de IAM que se crean como parte de la compilación de AMI personalizada.

Consulte [Modo PermissionsBoundary](#) para conocer la lista de políticas necesarias para utilizar dicha funcionalidad.

Configuraciones de red

AWS ParallelCluster utiliza Amazon Virtual Private Cloud (VPC) para las redes. VPC proporciona una plataforma de red flexible y configurable en la que puede implementar clústeres.

La VPC debe tener `DNS Resolution = yes`, `DNS Hostnames = yes` y opciones de DHCP con el nombre de dominio correcto para la región. El conjunto de opciones de DHCP predeterminado ya especifica los `AmazonProvidedDNS` necesarios. Si especifica más de un servidor de nombres de dominio, consulte [Conjuntos de opciones de DHCP](#) en la Guía del usuario de Amazon VPC.

AWS ParallelCluster admite las siguientes configuraciones de carácter general:

- Una subred para los nodos principal y de computación.
- Dos subredes: el nodo principal en una subred pública, y los nodos de computación en una subred privada. Las subredes pueden ser nuevas o existentes.

Todas estas configuraciones pueden funcionar con o sin direcciones IP públicas. AWS ParallelCluster también se puede implementar para usar un proxy HTTP para todas las solicitudes de AWS. La combinaciones de estas configuraciones se traducen en muchos casos de implementación. Por ejemplo, puede configurar una única subred pública con todos los accesos a través de Internet.

O bien, puede configurar una red totalmente privada mediante AWS Direct Connect y un proxy HTTP para todo el tráfico.

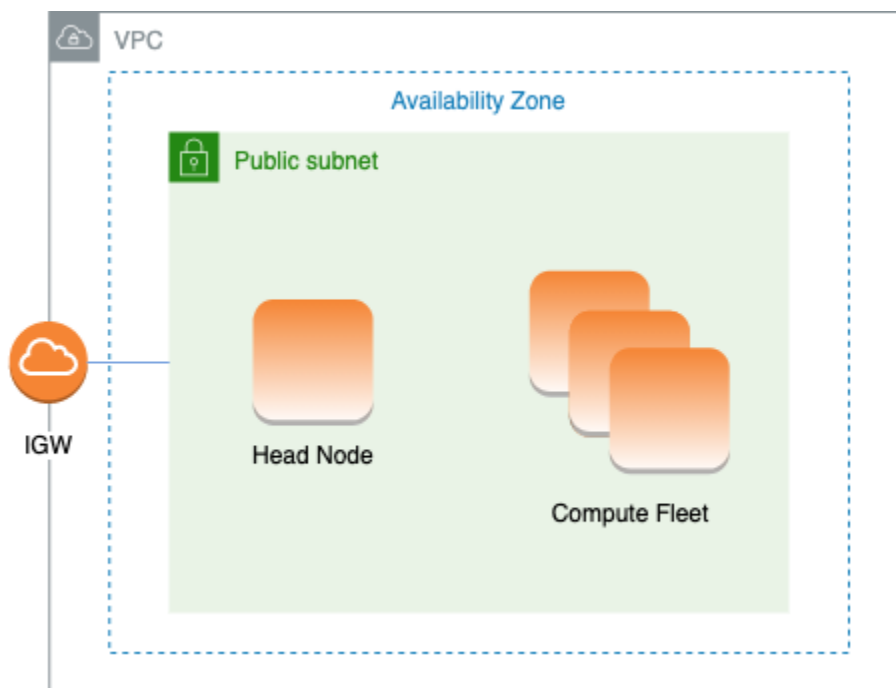
A partir de AWS ParallelCluster versión 3.0.0, es posible configurar distintos ajustes `SecurityGroups`, `AdditionalSecurityGroups` y `PlacementGroup` para cada cola. Para obtener más información, consulte [HeadNode/Networking](#) y [SlurmQueues/Networking](#) y [AwsBatchQueues/Networking](#).

Consulte los siguientes diagramas de arquitectura para ver ilustraciones de algunos de estos escenarios de redes.

Temas

- [AWS ParallelCluster en una subred pública única](#)
- [AWS ParallelCluster con dos subredes](#)
- [AWS ParallelCluster en una única subred privada conectada mediante AWS Direct Connect](#)
- [AWS ParallelCluster con el programador AWS Batch](#)
- [AWS ParallelCluster en una subred individual sin acceso a Internet](#)

AWS ParallelCluster en una subred pública única



La configuración de esta arquitectura requiere los siguientes valores de configuración:

```
# Note that all values are only provided as examples
HeadNode:
  ...
  Networking:
    SubnetId: subnet-12345678 # subnet with internet gateway
    #ElasticIp: true | false | eip-12345678
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - ...
    Networking:
      SubnetIds:
        - subnet-12345678 # subnet with internet gateway
      #AssignPublicIp: true
```

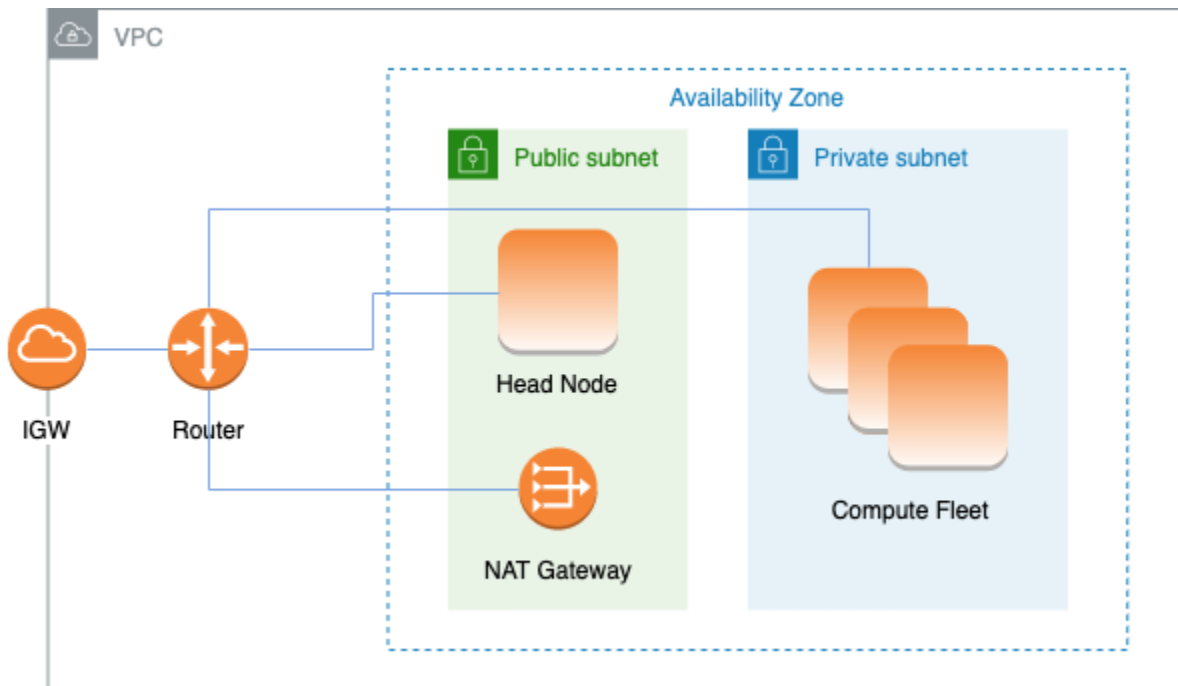
En esta configuración, a todas las instancias del clúster se les debe asignar una IP pública para poder acceder a Internet. Para ello, haga lo siguiente:

- Asegúrese de que al nodo principal se le asigne una dirección IP pública activando la configuración “Habilitar la asignación automática de direcciones IPv4 públicas” para la subred utilizada en [HeadNode/Networking/SubnetId](#) o asignando una IP elástica en [HeadNode/Networking/ElasticIp](#).
- Asegúrese de que a los nodos de computación se les asigne una dirección IP pública activando la configuración “Habilitar la asignación automática de direcciones IPv4 públicas” para la subred utilizada en [Scheduling/SlurmQueues/Networking/SubnetIds](#) o estableciendo [AssignPublicIp](#): verdadero en [Scheduling/SlurmQueues/Networking](#).
- Si define un tipo de instancia p4d u otro tipo de instancia que tenga varias interfaces de red o una tarjeta de interfaz de red para el nodo principal, debe configurar [HeadNode/Networking/ElasticIp](#) en true para proporcionar acceso público. Las IP públicas de AWS solo se pueden asignar a instancias lanzadas con una única interfaz de red. En este caso, le recomendamos que utilice una [puerta de enlace NAT](#) para proporcionar acceso público a los nodos de computación del clúster. Para obtener más información sobre las direcciones IP, consulte [Asignación de una dirección IPv4 pública durante el lanzamiento de una instancia](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.
- No puede definir un tipo de instancia p4d o hp6id, ni otro tipo de instancia que tenga varias interfaces de red o una tarjeta de interfaz de red para los nodos de computación, porque las IP públicas de AWS solo se pueden asignar a instancias lanzadas con una única interfaz de red. Para obtener más información sobre las direcciones IP, consulte [Asignación de una dirección](#)

[IPv4 pública durante el lanzamiento de una instancia](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

Para obtener más información, consulte [Habilitar acceso a Internet](#) en la Guía del usuario de Amazon VPC.

AWS ParallelCluster con dos subredes



La configuración para usar una subred privada existente para instancias de computación requiere los siguientes valores de configuración:

```
# Note that all values are only provided as examples
HeadNode:
  ...
  Networking:
    SubnetId: subnet-12345678 # subnet with internet gateway
    #ElasticIp: true | false | eip-12345678
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - ...
      Networking:
        SubnetIds:
          - subnet-23456789 # subnet with NAT gateway
```

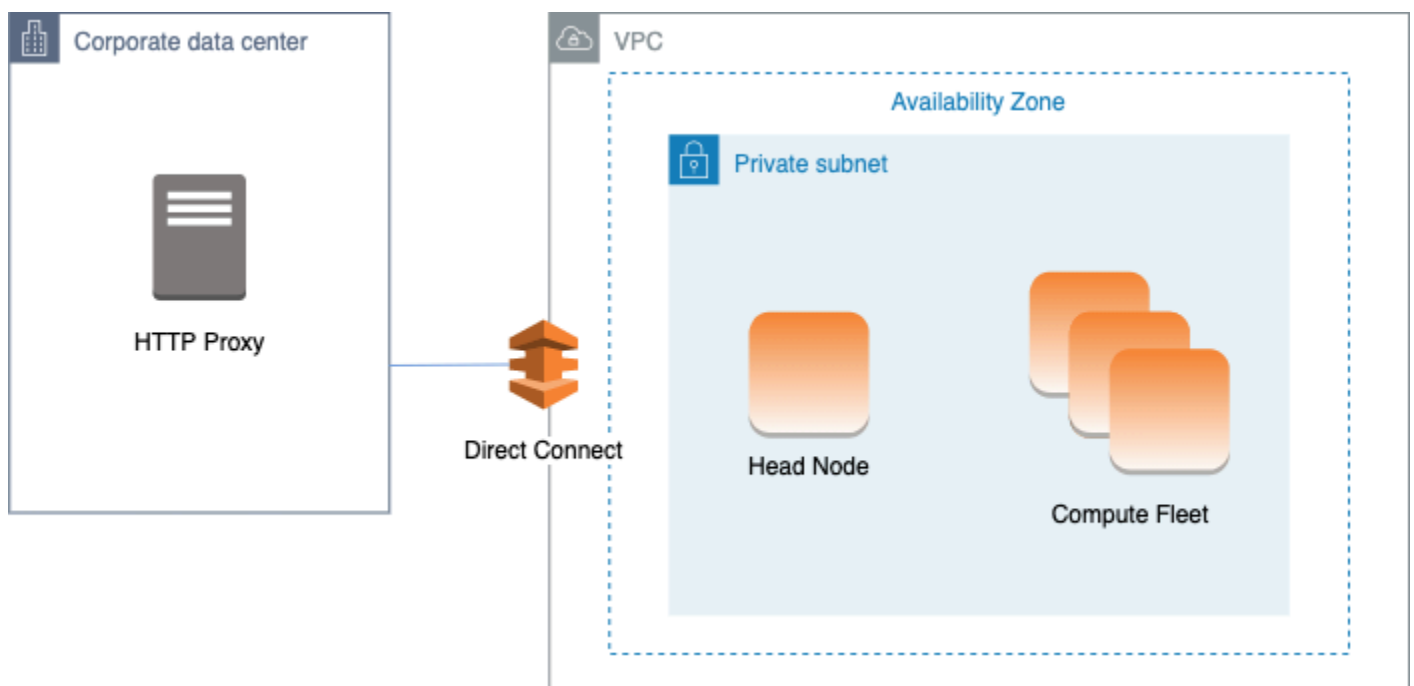
```
#AssignPublicIp: false
```

En esta configuración, solo se requiere que el nodo principal del clúster tenga asignada una IP pública. Puede hacerlo activando la configuración “Habilitar la asignación automática de direcciones IPv4 públicas” para la subred utilizada en [HeadNode/Networking/SubnetId](#) o asignando una IP elástica en [HeadNode/Networking/ElasticIp](#).

Si define un tipo de instancia p4d u otro tipo de instancia que tenga varias interfaces de red o una tarjeta de interfaz de red para el nodo principal, debe configurar [HeadNode/Networking/ElasticIp](#) en true para proporcionar acceso público. Las IP públicas de AWS solo se pueden asignar a instancias lanzadas con una única interfaz de red. Para obtener más información sobre las direcciones IP, consulte [Asignación de una dirección IPv4 pública durante el lanzamiento de una instancia](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

Esta configuración requiere una [puerta de enlace NAT](#) o un proxy interno en la subred utilizada para las colas, a fin de dar acceso a Internet a las instancias de computación.

AWS ParallelCluster en una única subred privada conectada mediante AWS Direct Connect



La configuración de esta arquitectura requiere los siguientes valores de configuración:

```
# Note that all values are only provided as examples
```

```
HeadNode:
  ...
  Networking:
    SubnetId: subnet-34567890 # subnet with proxy
    Proxy:
      HttpProxyAddress: http://proxy-address:port
  Ssh:
    KeyName: ec2-key-name
Scheduling:
  Scheduler: slurm
  SlurmQueues:
  - ...
    Networking:
      SubnetIds:
        - subnet-34567890 # subnet with proxy
      AssignPublicIp: false
      Proxy:
        HttpProxyAddress: http://proxy-address:port
```

Cuando se establece [Scheduling/SlurmQueues/Networking/AssignPublicIp](#) en false, las subredes deben configurarse correctamente para utilizar el proxy para todo el tráfico. Se requiere acceso a Internet tanto para los nodos principales como para los de computación.

AWS ParallelCluster con el programador AWS Batch

Al usar `awsbatch` como tipo de programador, AWS ParallelCluster crea un entorno informático administrado de AWS Batch. El entorno AWS Batch gestiona instancias de contenedores de Amazon Elastic Container Service (Amazon ECS). Estas instancias se lanzan en la subred configurada en el parámetro [AwsBatchQueues/Networking/SubnetIds](#). Para que AWS Batch funcione correctamente, las instancias de contenedor de Amazon ECS necesitan acceso de red externo para comunicarse con el punto de conexión de servicio de Amazon ECS. Esto se traduce en los siguientes casos:

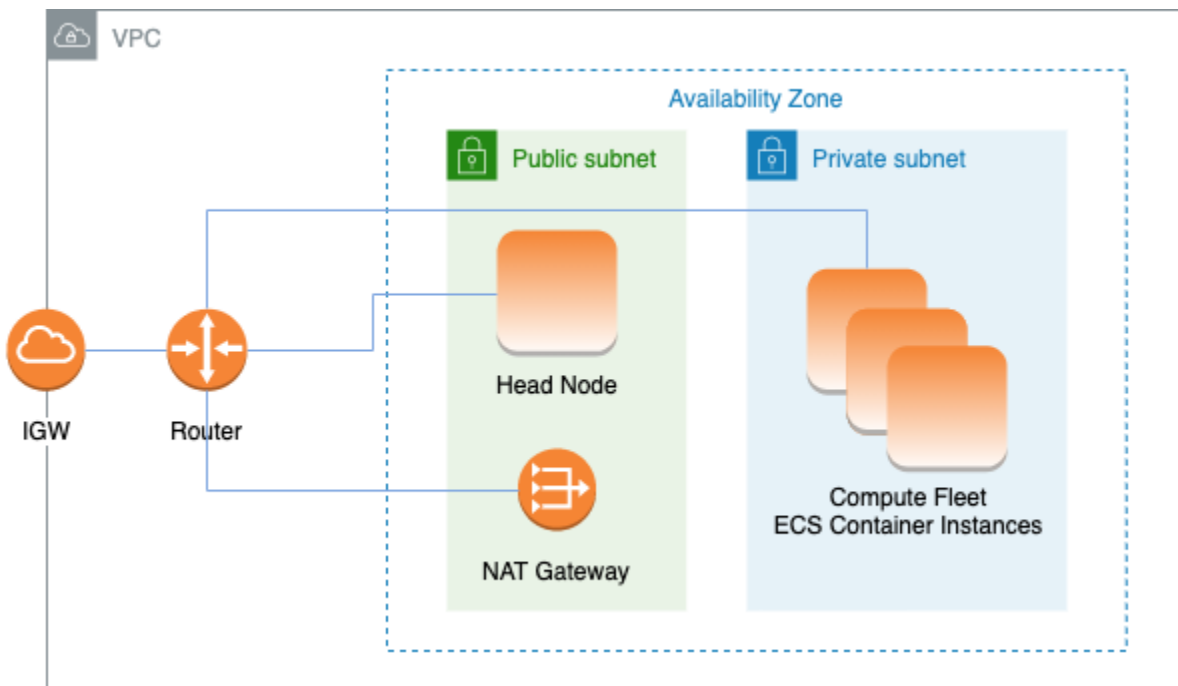
- El ID de subred especificado para la cola utiliza una [puerta de enlace NAT](#) para acceder a Internet. Recomendamos este enfoque.
- Las instancias que se lanzan en la subred de la cola tienen direcciones IP públicas y pueden llegar a Internet a través de una puerta de enlace de Internet.

Además, si le interesan los trabajos paralelos de varios nodos (de los [documentos de AWS Batch](#)):

Los trabajos paralelos de varios nodos de AWS Batch utilizan el modo de red `awsvpc` de Amazon ECS. Esto proporciona a los contenedores de trabajos paralelos de varios nodos las mismas propiedades de redes que poseen las instancias de Amazon EC2. Cada contenedor de trabajos paralelos de varios nodos obtiene su propia interfaz de red elástica, una dirección IP privada principal y un nombre de host DNS interno. La interfaz de red se crea en la misma subred de Amazon VPC como su recurso de computación de host. Los grupos de seguridad que se hayan aplicado a los recursos informáticos se aplicarán también a ella.

Cuando usa la integración de redes de tareas de Amazon ECS, el modo de red `awsvpc` no proporciona interfaces de red elásticas con direcciones IP públicas para tareas que utilizan el tipo de lanzamiento de Amazon EC2. Para obtener acceso a Internet, las tareas que utilizan el tipo de lanzamiento Amazon EC2 se deben lanzar en una subred privada configurada para utilizar una puerta de enlace NAT.

Debe configurar una [puerta de enlace NAT](#) para permitir que el clúster ejecute trabajos paralelos de varios nodos.



Todas las configuraciones y consideraciones anteriores también son válidas para AWS Batch. A continuación, se muestra un ejemplo de archivo de configuración de redes de AWS Batch.

```
# Note that all values are only provided as examples
HeadNode:
  ...
```

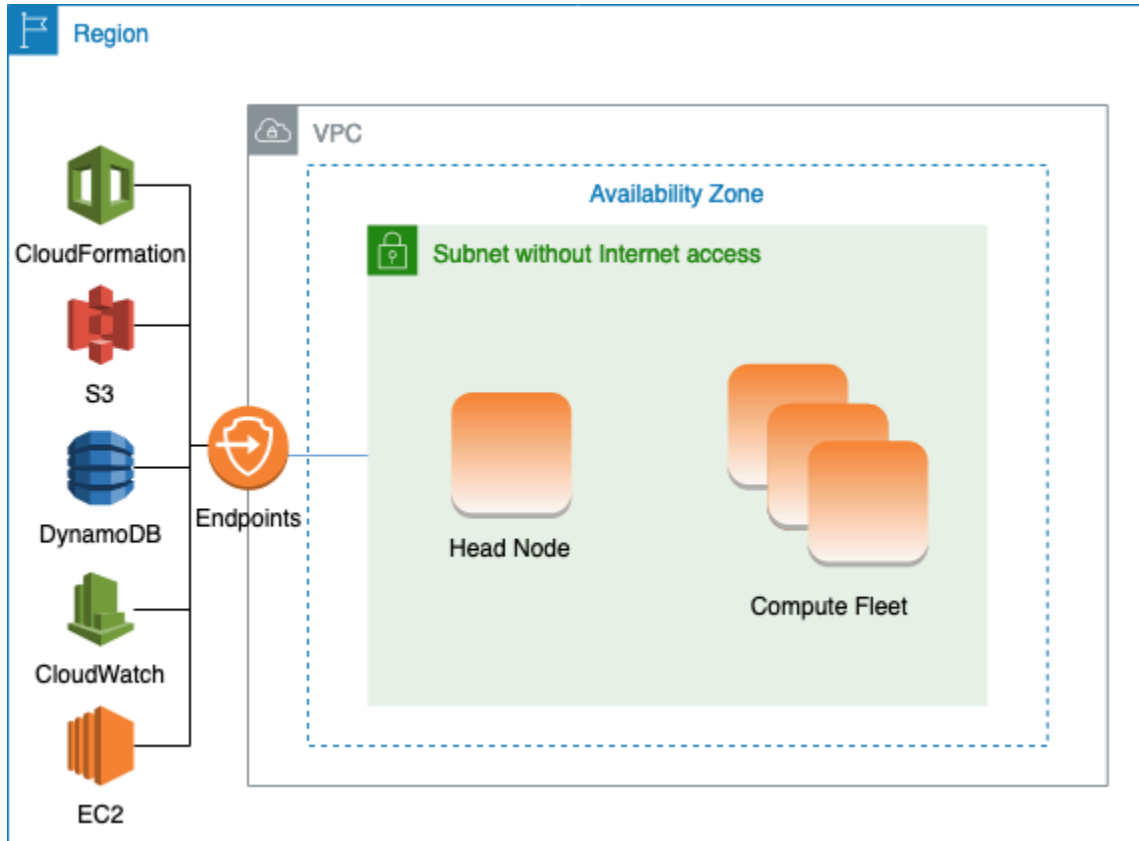
```
Networking:
  SubnetId: subnet-12345678 # subnet with internet gateway, NAT gateway or proxy
  #ElasticIp: true | false | eip-12345678
  #Proxy:
    #HttpProxyAddress: http://proxy-address:port
Ssh:
  KeyName: ec2-key-name
Scheduling:
  Scheduler: awsbatch
  AwsBatchQueues:
    - ...
    Networking:
      SubnetIds:
        - subnet-23456789 # subnet with internet gateway, NAT gateway or proxy
      #AssignPublicIp: true | false
```

En la sección [Scheduling/AwsBatchQueues/Networking](#), [SubnetIds](#) es un tipo de lista, pero actualmente solo se admite una subred.

Para obtener más información, consulte los siguientes temas:

- [Entornos de computación administrados de AWS Batch](#)
- [Trabajos paralelos de varios nodos de AWS Batch](#)
- [Integración de redes de tareas de Amazon ECS con el modo de red awsvpc](#)

AWS ParallelCluster en una subred individual sin acceso a Internet



Una subred sin acceso a Internet no permite conexiones entrantes ni salientes a Internet. Esta configuración de AWS ParallelCluster puede ayudar a los clientes preocupados por la seguridad a mejorar aún más la seguridad de sus recursos de AWS ParallelCluster. Los nodos de AWS ParallelCluster se crean a partir de AMI de AWS ParallelCluster que incluyen todo el software necesario para ejecutar un clúster sin acceso a Internet. De esta forma, AWS ParallelCluster puede crear y administrar clústeres con nodos que no tienen acceso a Internet.

En esta sección, aprenderá a configurar el clúster. También obtendrá información sobre las limitaciones de la ejecución de clústeres sin acceso a Internet.

Configuración de puntos de conexión de VPC

Para garantizar el correcto funcionamiento del clúster, los nodos del clúster deben poder interactuar con varios servicios de AWS.

Cree y configure los siguientes [puntos de conexión de VPC](#) para que los nodos del clúster puedan interactuar con los servicios de AWS sin acceso a Internet:

Commercial and AWS GovCloud (US) partitions

Servicio	Nombre del servicio	Tipo
Amazon CloudWatch	com.amazonaws. <i>region-id</i> .logs	Interfaz
AWS CloudFormation	com.amazonaws. <i>region-id</i> .cloudformation	Interfaz
Amazon EC2	com.amazonaws. <i>region-id</i> .ec2	Interfaz
Amazon S3	com.amazonaws. <i>region-id</i> .s3	Puerta de enlace
Amazon DynamoDB	com.amazonaws. <i>region-id</i> .dynamodb	Puerta de enlace
AWS Secrets Manager**	com.amazonaws. <i>region-id</i> .secretsmanager	Interfaz

China partition

Servicio	Nombre del servicio	Tipo
Amazon CloudWatch	com.amazonaws. <i>region-id</i> .logs	Interfaz
AWS CloudFormation	cn.com.amazonaws. <i>region-id</i> .cloudformation	Interfaz
Amazon EC2	cn.com.amazonaws. <i>region-id</i> .ec2	Interfaz
Amazon S3	com.amazonaws. <i>region-id</i> .s3	Puerta de enlace

Servicio	Nombre del servicio	Tipo
Amazon DynamoDB	com.amazonaws. <i>region-id</i> .dynamodb	Puerta de enlace
AWS Secrets Manager**	com.amazonaws. <i>region-id</i> .secretsmanager	Interfaz

** Este punto de conexión solo es obligatorio cuando está activado [DirectoryService](#); de lo contrario, es opcional.

Todas las instancias de la VPC deben tener los grupos de seguridad adecuados para comunicarse con los puntos de conexión. Para ello, agregue grupos de seguridad a [AdditionalSecurityGroups](#) en el [HeadNode](#) y los [AdditionalSecurityGroups](#) y en las configuraciones de las [SlurmQueues](#). Por ejemplo, si los puntos de conexión de VPC se crean sin especificar un grupo de seguridad de forma explícita, el grupo de seguridad predeterminado se asociará a los puntos de conexión. Al agregar el grupo de seguridad predeterminado a [AdditionalSecurityGroups](#), se habilita la comunicación entre el clúster y los puntos de conexión.

Note

Cuando utilice políticas de IAM para restringir el acceso a los puntos de conexión de VPC, debe añadir lo siguiente al punto de conexión de VPC de Amazon S3:

```
PolicyDocument:
  Version: 2012-10-17
  Statement:
    - Effect: Allow
      Principal: "*"
      Action:
        - "s3:PutObject"
      Resource:
        - !Sub "arn:${AWS::Partition}:s3::cloudformation-waitcondition-
          ${AWS::Region}/*"
```

Deshabilitar Route 53 y utilizar nombres de host de EC2

Al crear un clúster de Slurm, AWS ParallelCluster crea una zona alojada privada de Route 53 que se utiliza para resolver los nombres de host de los nodos de computación personalizados, por ejemplo `{queue_name}-{st|dy}-{compute_resource}-{N}`. Como Route 53 no admite puntos de conexión de VPC, esta característica debe estar deshabilitada. Además, AWS ParallelCluster debe configurarse para usar los nombres de host de EC2 predeterminados, como `ip-1-2-3-4`. Aplique los siguientes ajustes a la configuración del clúster:

```
...
Scheduling:
  ...
  SlurmSettings:
    Dns:
      DisableManagedDns: true
      UseEc2Hostnames: true
```

Warning

En el caso de los clústeres creados con los ajustes [SlurmSettings/Dns/DisableManagedDns](#) y [UseEc2Hostnames](#) establecidos en `true`, el DNS no resuelve el `nodeName` de Slurm. En su lugar, use el `NodeHostName` de Slurm.

Note

Esta nota no es pertinente a partir de AWS ParallelCluster versión 3.3.0.

Para las versiones de AWS ParallelCluster compatibles anteriores a 3.3.0:

Cuando `UseEc2Hostnames` se establece en `true`, el archivo de configuración de Slurm se configura con los scripts AWS ParallelCluster, `prolog` y `epilog`:

- `prolog` se ejecuta para añadir información de los nodos a `/etc/hosts` sobre los nodos de computación cuando se asigna cada trabajo.
- `epilog` se ejecuta para limpiar el contenido escrito por `prolog`.

Para añadir los scripts `prolog` o `epilog` personalizados, agréguelos a las carpetas `/opt/slurm/etc/pcluster/prolog.d/` o `/opt/slurm/etc/pcluster/epilog.d/`, respectivamente.

Configuración del clúster

Aprenda a configurar el clúster para que se ejecute en una subred sin conexión a Internet.

La configuración de esta arquitectura requiere los siguientes valores de configuración:

```
# Note that all values are only provided as examples
...
HeadNode:
  ...
  Networking:
    SubnetId: subnet-1234567890abcdef0 # the VPC of the subnet needs to have VPC
    endpoints
    AdditionalSecurityGroups:
      - sg-abcdef01234567890 # optional, the security group that enables the
    communication between the cluster and the VPC endpoints
Scheduling:
  Scheduler: slurm # Cluster in a subnet without internet access is supported only when
  the scheduler is Slurm.
  SlurmSettings:
    Dns:
      DisableManagedDns: true
      UseEc2Hostnames: true
  SlurmQueues:
    - ...
    Networking:
      SubnetIds:
        - subnet-1234567890abcdef0 # the VPC of the subnet needs to have VPC
    endpoints attached
      AdditionalSecurityGroups:
        - sg-1abcdef01234567890 # optional, the security group that enables the
    communication between the cluster and the VPC endpoints
```

- [SubnetId\(s\)](#): la subred sin acceso a Internet.

Para habilitar la comunicación entre los servicios AWS ParallelCluster y AWS, la VPC de la subred debe tener los puntos de conexión de VPC asociados. Antes de crear el clúster, compruebe que la [asignación automática de direcciones IPv4 públicas esté deshabilitada](#) en la subred para garantizar que los comandos de `pcluster` tengan acceso al clúster.

- [AdditionalSecurityGroups](#): el grupo de seguridad que habilita la comunicación entre el clúster y los puntos de conexión de VPC.

Opcional:

- Si los puntos de conexión de VPC se crean sin especificar un grupo de seguridad de forma explícita, se asociará el grupo de seguridad predeterminado de la VPC. Por lo tanto, proporcione el grupo de seguridad predeterminado a `AdditionalSecurityGroups`.
- Si se utilizan grupos de seguridad personalizados al crear el clúster o los puntos de conexión de VPC, no se necesita `AdditionalSecurityGroups`, siempre que los grupos de seguridad personalizados permitan la comunicación entre el clúster y los puntos de conexión de VPC.
- [Scheduler](#): el programador de clústeres.

`slurm` es el único valor válido. Solo el programador Slurm admite un clúster en una subred sin acceso a Internet.

- [SlurmSettings](#): la configuración de Slurm.

Consulte la sección anterior Deshabilitar Route 53 y utilizar los nombres de host de EC2.

Limitaciones

- Conexión al nodo principal a través de SSH o NICE DCV: al conectarse a un clúster, asegúrese de que el cliente de la conexión pueda llegar al nodo principal del clúster a través de su dirección IP privada. Si el cliente no está en la misma VPC que el nodo principal, use una instancia de proxy en una subred pública de la VPC. Este requisito se aplica a las conexiones SSH y DCV. No se puede acceder a la IP pública de un nodo principal si la subred no tiene acceso a Internet. Los comandos `pcluster ssh` y `dcv-connect` utilizan la IP pública, si existe, o la IP privada. Antes de crear el clúster, compruebe que la [asignación automática de direcciones IPv4 públicas esté deshabilitada](#) en la subred para garantizar que los comandos de `pcluster` tengan acceso al clúster.

El siguiente ejemplo muestra cómo puede conectarse a una sesión de DCV que se ejecute en el nodo principal del clúster. La conexión se realiza a través de una instancia de EC2 del proxy. La instancia funciona como un servidor NICE DCV para su PC y como cliente para el nodo principal de la subred privada.

Conéctese a través de DCV a través de una instancia proxy en una subred pública:

1. Cree una instancia de EC2 en una subred pública, que esté en la misma VPC que la subred del clúster.
2. Asegúrese de que el cliente y el servidor NICE DCV estén instalados en la instancia de EC2.

3. Asocie una política de usuario de AWS ParallelCluster a la instancia de EC2 del proxy. Para obtener más información, consulte [AWS ParallelCluster ejemplos pcluster de políticas de usuario](#).
 4. Instale AWS ParallelCluster en la instancia de EC2 del proxy.
 5. Conéctese a través de DCV a la instancia de EC2 del proxy.
 6. Utilice el comando `pcluster dcv-connect` de la instancia del proxy para conectarse al clúster dentro de la subred sin acceso a Internet.
- Interacción con otros servicios de AWS: arriba solo figuran los servicios estrictamente necesarios para AWS ParallelCluster. Si el clúster debe interactuar con otros servicios, cree los puntos de conexión de VPC correspondientes.

Nodos de inicio de sesión

A partir de la versión 3.7.0, los administradores de clústeres de AWS ParallelCluster pueden aprovisionar nodos de inicio de sesión que se pueden utilizar para proporcionar acceso a los usuarios para ejecutar trabajos, en lugar de acceder directamente al nodo principal del clúster. Los usuarios del clúster con los permisos adecuados pueden usar Active Directory o su credencial SSH para iniciar sesión y para enviar y administrar sus trabajos. Como resultado, se puede mejorar la administración de clústeres y se pueden minimizar las posibilidades de agotar los recursos del nodo principal que Slurm necesita para administrar el clúster. Los usuarios que hayan iniciado sesión también tendrán acceso a todo el almacenamiento compartido del clúster montado en los nodos de inicio de sesión. Si es necesario detener la sesión de un nodo de inicio de sesión, se notificará previamente a los usuarios a través de la sesión de shell activa que estén utilizando.

Los nodos de inicio de sesión se especifican como grupos, donde un grupo define un grupo de nodos de inicio de sesión que tienen la misma configuración de recursos. Todos los nodos de inicio de sesión de un grupo están configurados para formar parte de un [equilibrador de carga de red](#) que permite distribuir las sesiones entre los nodos de inicio de sesión de forma sucesiva. La presente implementación permite especificar un grupo de nodos de inicio de sesión que contiene varios nodos de inicio de sesión.

Seguridad

Los nodos de inicio de sesión heredan la configuración de AllowedIPs [AllowedIps](#) del nodo principal. De esta forma, los administradores del clúster pueden restringir la seguridad del clúster

especificando el CIDR de origen o una lista de prefijos desde la que se permiten las conexiones SSH.

En la presente implementación, el acceso al nodo principal no se restringe automáticamente al habilitar los nodos de inicio de sesión. Si es necesario, un administrador de clústeres puede restringir este acceso actualizando la configuración SSH del nodo principal mediante comandos estándar de Linux. Esto también se puede lograr especificando grupos de seguridad personalizados en el nodo principal mediante la configuración de `AdditionalSecurityGroups` en la sección del nodo principal del archivo YAML de ParallelCluster para denegar las conexiones de usuarios no autorizados.

Redes

Los nodos de inicio de sesión se aprovisionan con una única dirección de conexión al equilibrador de carga de red configurado para el grupo de nodos de inicio de sesión. La configuración de conectividad de la dirección se basa en el tipo de subred especificado en la configuración del grupo de nodos de inicio de sesión.

- Si la subred es privada, la dirección será privada y, para permitir el acceso a los nodos de inicio de sesión, el administrador del clúster debe aprovisionar un host bastión.
- Si la subred es pública, la dirección será pública.

El equilibrador de carga de red administra todas las solicitudes de conexión mediante el enrutamiento por turnos.

Almacenamiento

Todo el almacenamiento compartido configurado en el clúster mediante ParallelCluster, incluido el almacenamiento administrado, se montará en todos los nodos de inicio de sesión.

Recuperación de la información de los nodos de inicio de sesión

Para recuperar la dirección de la conexión única aprovisionada para acceder a los nodos de inicio de sesión, el administrador del clúster puede ejecutar el comando [describe-cluster](#). El comando también proporcionará más información sobre el estado de los nodos de inicio de sesión.

Los nodos de inicio de sesión son un nuevo tipo de nodo compatible con ParallelCluster que se puede especificar con el comando [describe-cluster-instances](#) al consultar el estado de un tipo de nodo específico.

La disponibilidad de una dirección de conexión única para el grupo de nodos de inicio de sesión no impide el acceso directo a un nodo de inicio de sesión específico. Sin embargo, no se recomienda utilizar la conexión directa para evitar las advertencias del cliente SSH. El cliente SSH almacena los identificadores de host de forma local para cada dirección de destino. Como el identificador de host es específico de cada grupo, el uso de diferentes direcciones IP o de una sola dirección de conexión puede tener el mismo identificador de host asociado a diferentes direcciones de destino, lo que puede provocar una advertencia por parte del cliente SSH, ya que el mismo identificador de host está asociado a varios destinos.

Propiedades del IMDS

El acceso al IMDS del nodo de inicio de sesión (y a las credenciales del perfil de la instancia) está restringido al usuario raíz, al usuario administrador del clúster (`pc-cluster-admin` de forma predeterminada) y al usuario predeterminado específico del sistema operativo (`ec2-user` en Amazon Linux 2 y RedHat, `ubuntu` en Ubuntu 18.04 y `centos` en CentOS 7).

Para restringir el acceso al IMDS, AWS ParallelCluster administra una cadena de `iptables`.

Note

Cualquier personalización de las reglas `iptables` o `ip6tables` puede interferir con el mecanismo utilizado para restringir el acceso al IMDS en el nodo de inicio de sesión. Consulte también [Imds property setting](#).

Ciclo de vida de los nodos de inicio de sesión

Actualmente, no hay ningún comando dedicado a detener e iniciar los nodos de inicio de sesión de un grupo. Para detener los nodos de inicio de sesión de un grupo, el administrador del clúster debe actualizar la configuración del clúster especificando un cero en el recuento de nodos de inicio de sesión (`Count: 0`) y, a continuación, ejecutar un comando [pcluster.update-cluster-v3](#).

Note

Los usuarios que han iniciado sesión reciben una notificación sobre la finalización de la instancia específica y sobre el periodo de gracia correspondiente. Durante el periodo de gracia, no se permitirán nuevas conexiones, excepto las del [usuario predeterminado del clúster](#). El administrador del clúster puede personalizar el mensaje que se muestra

desde el nodo principal o desde un nodo de inicio de sesión que edita el archivo `/opt/parallelcluster/shared_login_nodes/loginmgt_config.json`.

Para iniciar el grupo de nodos de inicio de sesión, el administrador del clúster debe restaurar el valor Count anterior de la configuración del clúster y, a continuación, ejecutar un comando [update-cluster](#).

Permisos necesarios para ejecutar el grupo de nodos de inicio de sesión

Para administrar el grupo de nodos de inicio de sesión, el administrador del clúster debe tener los siguientes permisos adicionales:

```
- Action:
  - autoscaling:DeleteAutoScalingGroup
  - autoscaling:DeleteLifecycleHook
  - autoscaling:Describe*
  - autoscaling:PutLifecycleHook
  - autoscaling:UpdateAutoScalingGroup
  - elasticloadbalancing:CreateListener
  - elasticloadbalancing:CreateTargetGroup
  - elasticloadbalancing>DeleteListener
  - elasticloadbalancing>DeleteLoadBalancer
  - elasticloadbalancing>DeleteTargetGroup
  - elasticloadbalancing:Describe*
  - elasticloadbalancing:ModifyLoadBalancerAttributes
Resource: '*'
Condition:
  ForAllValues:StringEquals:
    aws:TagKeys: [ "parallelcluster:cluster-name" ]
- Action:
  - autoscaling:CreateAutoScalingGroup
  - elasticloadbalancing:AddTags
  - elasticloadbalancing:CreateLoadBalancer
Resource: '*'
Effect: Allow
```

Acciones de arranque personalizadas

Si se definen los ajustes de configuración [HeadNode/CustomActions/OnNodeStart](#), AWS ParallelCluster ejecuta código arbitrario inmediatamente después de que se inicie el nodo. Si se definen los ajustes de configuración [HeadNode/CustomActions/OnNodeConfigured](#), AWS ParallelCluster ejecuta el código después de que finalice correctamente la configuración del nodo.

A partir de AWS ParallelCluster versión 3.4.0, el código se puede ejecutar después de la actualización del nodo principal, si se definen los ajustes de configuración [HeadNode/CustomActions/OnNodeUpdated](#).

En la mayoría de los casos, este código se almacena en Amazon Simple Storage Service (Amazon S3) y se accede a él a través de una conexión HTTPS. El código se ejecuta como `root` y puede estar en cualquier lenguaje de script compatible con el sistema operativo del clúster. A menudo, el código está en Bash o Python.

Note

A partir de AWS ParallelCluster versión 3.7.0, la configuración predeterminada de los ajustes [Imds/ImdsSupport](#) del clúster es `v2.0`.

Al crear un clúster nuevo para actualizarlo a la versión 3.7.0 y versiones posteriores, actualice los scripts de acción de arranque personalizados para que sean compatibles con IMDSv2 o defina [Imds/ImdsSupport](#) como `v1.0` en el archivo de configuración del clúster.

Warning

Usted es responsable de configurar los scripts y argumentos personalizados tal y como se describe en el [modelo de responsabilidad compartida](#). Compruebe que los scripts y argumentos de arranque personalizados provengan de fuentes en las que confía para tener acceso total a los nodos del clúster.

⚠ Warning

AWS ParallelCluster no admite el uso de variables internas que se proporcionan a través del archivo `/etc/parallelcluster/cfnconfig`. Es posible que este archivo se elimine como parte de una versión futura.

Las acciones `OnNodeStart` se invocan antes de que se inicie cualquier acción de arranque de implementación de nodos, como la configuración de NAT, Amazon Elastic Block Store (Amazon EBS) o el programador. Las acciones de arranque `OnNodeStart` pueden incluir la modificación del almacenamiento, la adición de usuarios adicionales y la adición de paquetes.

ℹ Note

Si configura un [DirectoryService](#) y un script [HeadNode/CustomActions/OnNodeStart](#) para el clúster, AWS ParallelCluster configura `DirectoryService` y reinicia el `sssd`, antes de ejecutar el script `OnNodeStart`.

Las acciones `OnNodeConfigured` se invocan una vez finalizados los procesos de arranque del nodo. Las acciones `OnNodeConfigured` son las últimas acciones que se producen antes de que se considere que una instancia está totalmente configurada y completa. Algunas acciones `OnNodeConfigured` incluyen la modificación de la configuración del programador, del almacenamiento o de los paquetes. Puede pasar argumentos a los scripts especificándolos durante la configuración.

Las acciones `OnNodeUpdated` se ejecutan cuando se completa la actualización del nodo principal y cuando el programador y el almacenamiento compartido se ajustan a los cambios más recientes en la configuración del clúster.

Cuando las acciones personalizadas `OnNodeStart` o `OnNodeConfigured` se completan correctamente, se indica con el código de salida cero (0). Cualquier otro código de salida indica que se ha producido un error en el arranque de la instancia.

Cuando las acciones personalizadas `OnNodeUpdated` se completan correctamente, se señala con el código de salida cero (0). Cualquier otro código de salida indica que se ha producido un error en la actualización.

Note

Si configura [OnNodeUpdated](#), debe restaurar manualmente las `OnNodeUpdated` acciones al estado anterior en caso de que se produzcan errores de actualización.

Si se produce un error en una acción personalizada `OnNodeUpdated`, la actualización vuelve al estado anterior. Sin embargo, la acción `OnNodeUpdated` solo se ejecuta en el momento de la actualización y no en el momento de la reversión de la pila.

Puede especificar scripts diferentes para el nodo principal y para cada cola en las secciones de configuración [HeadNode/CustomActions](#) y [Scheduling/SlurmQueues/CustomActions](#). [OnNodeUpdated](#) solo se puede configurar en la sección `HeadNode`.

Note

Antes de AWS ParallelCluster versión 3.0, no era posible especificar scripts diferentes para los nodos principal y de computación. Consulte [Transición de AWS ParallelCluster 2.x a 3.x](#).

Temas

- [Configuración](#)
- [Arguments](#)
- [Ejemplo de clúster con acciones de arranque personalizadas](#)
- [Ejemplo de actualización de un script de arranque personalizado para IMDSv2](#)
- [Ejemplo de actualización de una configuración para IMDSv1](#)

Configuración

Los siguientes ajustes de configuración se utilizan para definir las acciones y argumento [HeadNode/CustomActions/OnNodeStart](#) y [OnNodeConfigured](#) y [OnNodeUpdated](#) y [Scheduling/CustomActions/OnNodeStart](#) y [OnNodeConfigured](#).

```
HeadNode:
  [...]
  CustomActions:
    OnNodeStart:
      # Script URL. This is run before any of the bootstrap scripts are run
```

```
  Script: s3://bucket-name/on-node-start.sh
  Args:
    - arg1
  OnNodeConfigured:
    # Script URL. This is run after all the bootstrap scripts are run
    Script: s3://bucket-name/on-node-configured.sh
    Args:
      - arg1
  OnNodeUpdated:
    # Script URL. This is run after the head node update is completed.
    Script: s3://bucket-name/on-node-updated.sh
    Args:
      - arg1
# Bucket permissions
Iam:
  S3Access:
    - BucketName: bucket_name
      EnableWriteAccess: false
Scheduling:
  Scheduler: slurm
  [...]
SlurmQueues:
  - Name: queue1
  [...]
CustomActions:
  OnNodeStart:
    Script: s3://bucket-name/on-node-start.sh
    Args:
      - arg1
  OnNodeConfigured:
    Script: s3://bucket-name/on-node-configured.sh
    Args:
      - arg1
  Iam:
    S3Access:
      - BucketName: bucket_name
        EnableWriteAccess: false
```

Uso del ajuste Sequence (agregado en AWS ParallelCluster versión 3.6.0):

```
HeadNode:
  [...]
  CustomActions:
```



```
OnNodeStart:
  # Script URLs. The scripts are run in the same order as listed in the
  # configuration, before any of the bootstrap scripts are run.
  Sequence:
    - Script: s3://bucket-name/on-node-start1.sh
      Args:
        - arg1
    - Script: s3://bucket-name/on-node-start2.sh
      Args:
        - arg1
    [...]
OnNodeConfigured:
  # Script URLs. The scripts are run in the same order as listed in the
  # configuration, after all the bootstrap scripts are run.
  Sequence:
    - Script: s3://bucket-name/on-node-configured1.sh
      Args:
        - arg1
    - Script: s3://bucket-name/on-node-configured2.sh
      Args:
        - arg1
    [...]
OnNodeUpdated:
  # Script URLs. The scripts are run in the same order as listed in the
  # configuration, after the head node update is completed.
  Sequence:
    - Script: s3://bucket-name/on-node-updated1.sh
      Args:
        - arg1
    - Script: s3://bucket-name/on-node-updated2.sh
      Args:
        - arg1
    [...]
# Bucket permissions
Iam:
  S3Access:
    - BucketName: bucket_name
      EnableWriteAccess: false
Scheduling:
  Scheduler: slurm
  [...]
SlurmQueues:
  - Name: queue1
  [...]
```

```

CustomActions:
  OnNodeStart:
    # Script URLs. The scripts are run in the same order as listed in the
    configuration, before any of the bootstrap scripts are run
    Sequence:
      - Script: s3://bucket-name/on-node-start1.sh
        Args:
          - arg1
      - Script: s3://bucket-name/on-node-start2.sh
        Args:
          - arg1
      [...]
  OnNodeConfigured:
    # Script URLs. The scripts are run in the same order as listed in the
    configuration, after all the bootstrap scripts are run
    Sequence:
      - Script: s3://bucket-name/on-node-configured1.sh
        Args:
          - arg1
      - Script: s3://bucket-name/on-node-configured2.sh
        Args:
          - arg1
      [...]
  Iam:
    S3Access:
      - BucketName: bucket_name
        EnableWriteAccess: false

```

El ajuste Sequence se añade a partir de AWS ParallelCluster versión 3.6.0. Si especifica Sequence, puede mostrar varios scripts para una acción personalizada. AWS ParallelCluster sigue admitiendo la configuración de una acción personalizada con un único script, sin incluir Sequence.

AWS ParallelCluster no admite incluir tanto un único script como Sequence para la misma acción personalizada. Por ejemplo, AWS ParallelCluster muestra un error si especifica la siguiente configuración.

```

[...]
CustomActions:
  OnNodeStart:
    # Script URL. This is run before any of the bootstrap scripts are run
    Script: s3://bucket-name/on-node-start.sh
    Args:
      - arg1

```

```
# Script URLs. The scripts are run in the same order as listed in the
configuration, before any of the bootstrap scripts are run.
```

```
Sequence:
```

```
- Script: s3://bucket-name/on-node-start1.sh
```

```
  Args:
```

```
    - arg1
```

```
- Script: s3://bucket-name/on-node-start2.sh
```

```
  Args:
```

```
    - arg1
```

```
[...]
```

Arguments

Note

En la AWS ParallelCluster 2.x, los argumentos \$1 estaban reservados para almacenar la URL del script personalizado. Si quiere reutilizar los scripts de arranque personalizados creados para la AWS ParallelCluster 2.x con la AWS ParallelCluster 3.x, debe adaptarlos teniendo en cuenta el cambio de los argumentos. Consulte [Transición de AWS ParallelCluster 2.x a 3.x](#).

Ejemplo de clúster con acciones de arranque personalizadas

Los siguientes pasos crean un script simple que se ejecuta después de configurar el nodo y que instala los paquetes `R`, `curl` y `wget` en los nodos del clúster.

1. Cree un script.

```
#!/bin/bash
echo "The script has $# arguments"
for arg in "$@"
do
    echo "arg: ${arg}"
done
yum -y install "${@:1}"
```

2. Cargue el script con los permisos correctos en Amazon S3. Si los permisos de lectura públicos no son adecuados para usted, utilice las secciones de configuración [HeadNode/Iam/S3Access](#) y [Scheduling/SlurmQueues](#). Para más información, consulte [Uso de Amazon S3](#).

```
$ aws s3 cp --acl public-read /path/to/myscript.sh s3://<bucket-name>/myscript.sh
```

⚠ Important

Si la secuencia de comandos se editó en Windows, los finales de línea deben cambiarse de CRLF a LF antes de cargar la secuencia de comandos en Amazon S3.

3. Actualice la configuración de AWS ParallelCluster para incluir la nueva acción `OnNodeConfigured`.

```
CustomActions:
OnNodeConfigured:
  Script: https://<bucket-name>.s3.<region>.amazonaws.com/myscript.sh
  Args:
    - "R"
    - "curl"
    - "wget"
```

Si el bucket no tiene permiso de lectura pública, utilice `s3` como protocolo de la URL.

```
CustomActions:
OnNodeConfigured:
  Script: s3://<bucket-name>/myscript.sh
  Args:
    - "R"
    - "curl"
    - "wget"
```

4. Lance el clúster.

```
$ pcluster create-cluster --cluster-name mycluster \
  --region <region> --cluster-configuration config-file.yaml
```

5. Verifique el resultado.

- Si ha agregado acciones personalizadas a la configuración de `HeadNode`, inicie sesión en el nodo principal y compruebe el archivo `cfn-init.log` ubicado en `/var/log/cfn-init.log` ejecutando el siguiente comando:

```
$ less /var/log/cfn-init.log
```

```
2021-09-03 10:43:54,588 [DEBUG] Command run
postinstall output: The script has 3 arguments
arg: R
arg: curl
arg: wget
Loaded plugins: dkms-build-requires, priorities, update-motd, upgrade-helper
Package R-3.4.1-1.52.amzn1.x86_64 already installed and latest version
Package curl-7.61.1-7.91.amzn1.x86_64 already installed and latest version
Package wget-1.18-4.29.amzn1.x86_64 already installed and latest version
Nothing to do
```

- Si ha agregado acciones personalizadas a la configuración de SlurmQueues, compruebe `cloud-init.log` ubicado en `/var/log/cloud-init.log` en un nodo de computación. Se utiliza CloudWatch para ver estos registros.

Puedes ver ambos registros en la CloudWatch consola de Amazon. Para más información, consulte [Integración con Amazon CloudWatch Logs](#).

Ejemplo de actualización de un script de arranque personalizado para IMDSv2

En el siguiente ejemplo, actualizamos un script de acción de arranque personalizado que se usó con IMDSv1 para usarlo con IMDSv2. El script IMDSv1 recupera los metadatos del identificador de AMI de la instancia de EC2.

```
#!/bin/bash
AMI_ID=$(curl http://169.254.169.254/latest/meta-data/ami-id)
echo $AMI_ID >> /home/ami_id.txt
```

A continuación se muestra el script de acción de arranque personalizado modificado para que sea compatible con IMDSv2.

```
#!/bin/bash
AMI_ID=$(TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600" ` \
    && curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/
latest/meta-data/ami-id)
echo $AMI_ID >> /home/ami_id.txt
```

Para obtener más información, consulte [Recuperación de metadatos de instancia](#) en la Guía del usuario de EC2 para instancias de Linux.

Ejemplo de actualización de una configuración para IMDSv1

El siguiente es un ejemplo de una configuración de clúster que admite IMDSv1 cuando se utiliza AWS ParallelCluster versión 3.7.0 y anteriores.

```
Region: us-east-1
Imds:
  ImdsSupport: v1.0
Image:
  Os: alinux2
HeadNode:
  InstanceType: t2.micro
  Networking:
    SubnetId: subnet-abcdef01234567890
  Ssh
    KeyName: key-name
  CustomActions:
    OnNodeConfigured:
      Script: Script-path
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue1
      CustomActions:
        OnNodeConfigured:
          Script: Script-path
  ComputeResources:
    - Name: t2micro
      Instances:
        - InstanceType: t2.micro
      MinCount: 11
  Networking:
    SubnetIds:
      - subnet-abcdef01234567890
```

Uso de Amazon S3

Puede configurar el acceso de AWS ParallelCluster a Amazon S3 a través de los parámetros [HeadNode/Iam/S3Access](#) y [Scheduling/SlurmQueues/- Name/Iam/S3Access](#) en la configuración de AWS ParallelCluster.

Ejemplos

El siguiente ejemplo configura el acceso de solo lectura a todos los objetos de entrada en *firstbucket/read_only/* y el acceso de lectura/escritura a todos los objetos de entrada en *secondbucket/read_and_write/*.

```
...
HeadNode:
  ...
  Iam:
    S3Access:
      - BucketName: firstbucket
        KeyName: read_only/*
        EnableWriteAccess: false
      - BucketName: secondbucket
        KeyName: read_and_write/*
        EnableWriteAccess: true
  ...
```

El siguiente ejemplo configura el acceso de solo lectura a todos los objetos de la carpeta *read_only/* de cualquier bucket (*) de la cuenta.

```
...
HeadNode:
  ...
  Iam:
    S3Access:
      - BucketName: *
        KeyName: read_only/*
        EnableWriteAccess: false
  ...
```

El ejemplo final configura el acceso de solo lectura a todos los buckets y objetos de la cuenta.

```
...
```

```
HeadNode:
  ...
  Iam:
    S3Access:
      - BucketName: *
  ...
```

Uso de instancias de spot

AWS ParallelCluster usa instancias puntuales si ha establecido [SlurmQueues/CapacityType](#) o [AwsBatchQueues/CapacityTypeSP0T](#) en el archivo de configuración del clúster. Las instancias de spot son más rentables que las instancias bajo demanda, pero pueden interrumpirse. Puede resultar útil utilizar los avisos de interrupción de instancias de spot, que envían una advertencia dos minutos antes de que Amazon EC2 tenga que detener o terminar su instancia de spot. Para obtener más información, consulte [Interrupciones de instancias puntuales](#) en la Guía del usuario de Amazon EC2. Para saber cómo [AwsBatchQueues](#) funciona con las instancias de spot, consulte [Recursos de cómputo](#) en la Guía del usuario AWS Batch .

El programador AWS ParallelCluster configurado asigna los trabajos a los recursos de cómputo de las colas con instancias puntuales del mismo modo que asigna los trabajos a los recursos de cómputo de las colas con instancias bajo demanda.

Al utilizar instancias puntuales, debe existir un rol `AWSServiceRoleForEC2Spot` vinculado a un servicio en su cuenta. Para crear este rol en su cuenta mediante el AWS CLI, ejecute el siguiente comando:

```
$ aws iam create-service-linked-role --aws-service-name spot.amazonaws.com
```

Para obtener más información, consulte [Función vinculada a servicios para solicitudes de instancias puntuales](#) en la Guía del usuario de Amazon EC2.

En las secciones siguientes se describen tres escenarios en los que las instancias de spot pueden interrumpirse al usar [SlurmQueues](#).

Escenario 1: Se interrumpe una instancia de spot sin trabajos en ejecución

Cuando se produzca esta interrupción, AWS ParallelCluster intenta reemplazar la instancia si la cola del programador tiene trabajos pendientes que requieren instancias adicionales o si el número

de instancias activas es inferior a//. [SlurmQueuesComputeResourcesMinCount](#) Si no AWS ParallelCluster se pueden aprovisionar nuevas instancias, se repite periódicamente la solicitud de nuevas instancias.

Escenario 2: Se interrumpe una instancia de spot que ejecuta trabajos de un solo nodo

El trabajo falla con un código de `NODE_FAIL` estado igual a `y` y se vuelve a poner en cola (a menos que `--no-requeue` se especifique al enviar el trabajo). Si el nodo es estático, se reemplaza. Si el nodo es un nodo dinámico, se termina y se restablece. Para obtener más información sobre `sbatch`, incluido el parámetro `--no-requeue`, incluido este, consulte [sbatch](#) en la documentación de Slurm.

Escenario 3: Se interrumpe una instancia de spot que ejecuta trabajos de varios nodos

El trabajo falla con un código de `NODE_FAIL` estado igual a `y` y se vuelve a poner en cola (a menos que `--no-requeue` se especifique al enviar el trabajo). Si el nodo es estático, se reemplaza. Si el nodo es un nodo dinámico, se termina y se restablece. Otros nodos que estaban ejecutando los trabajos terminados se pueden asignar a otros trabajos pendientes o reducirse verticalmente una vez transcurrido el tiempo especificado en [SlurmSettings/ScaledownIdleTime](#).

Para obtener más información sobre las instancias puntuales, consulte [las instancias puntuales](#) en la Guía del usuario de Amazon EC2.

Programadores compatibles con AWS ParallelCluster

Programadores compatibles con AWS ParallelCluster

AWS ParallelCluster soporta Slurm y AWS Batch programadores, configurados mediante la configuración. [Scheduler](#)

Temas

- [Slurm Workload Manager \(slurm\)](#)
- [AWS Batch \(awsbatch\)](#)

Slurm Workload Manager (**slurm**)

Capacidad, tamaño y actualización del clúster

La capacidad del clúster se define por la cantidad de nodos de procesamiento que el clúster puede escalar. Los nodos de cómputo están respaldados por instancias de EC2 definidas dentro de los recursos de cómputo de la AWS ParallelCluster configuración ([Scheduling/SlurmQueues/ComputeResources](#)) y están organizados en colas ([Scheduling/SlurmQueues](#)) que se asignan a Slurm las particiones 1:1.

Dentro de un recurso de cómputo, es posible configurar el número mínimo de nodos de cómputo (instancias) que deben mantenerse siempre en ejecución en el clúster ([MinCount](#)) y el número máximo de instancias al que el recurso de cómputo puede escalar ([MaxCount3](#)).

En el momento de la creación del clúster, o tras una actualización del clúster, AWS ParallelCluster lanza tantas instancias de EC2 como estén configuradas `MinCount` para cada recurso informático ([Scheduling/SlurmQueues/ ComputeResources](#)) definido en el clúster. Las instancias lanzadas para cubrir la cantidad mínima de nodos de un recurso informático del clúster se denominan nodos estáticos. Una vez iniciados, los nodos estáticos deben permanecer en el clúster y el sistema no los termina, a menos que se produzca un evento o una condición en particular. Estos eventos incluyen, por ejemplo, el fallo de las comprobaciones de estado de EC2 y el cambio del estado del nodo Slurm a DRAIN Slurm o DOWN.

Las instancias de EC2, **1** que `'MaxCount - MinCount'` se lanzan bajo demanda para hacer frente al aumento de carga del clúster, se denominan nodos dinámicos. **MaxCount MinCount**) Su naturaleza es efímera: se lanzan para atender tareas pendientes y se cancelan una vez que permanecen inactivas durante un período de tiempo definido [Scheduling/SlurmSettings/ScaledownIdleTime](#) en la configuración del clúster (predeterminado: 10 minutos).

Los nodos estáticos y los nodos dinámicos cumplen con el siguiente esquema de nomenclatura:

- Nodos estáticos `<Queue/Name>-st-<ComputeResource/Name>-<num>` donde `<num> = 1..ComputeResource/MinCount`
- `<Queue/Name>-dy-<ComputeResource/Name>-<num>` ¿Dónde están los nodos dinámicos? `<num> = 1..(ComputeResource/MaxCount - ComputeResource/MinCount)`

Por ejemplo, dada la siguiente AWS ParallelCluster configuración:

```
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue1
      ComputeResources:
        - Name: c5xlarge
          Instances:
            - InstanceType: c5.xlarge
              MinCount: 100
              MaxCount: 150
```

Los siguientes nodos se definirán en Slurm

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up        infinite   50    idle~ queue1-dy-c5xlarge-[1-50]
queue1*    up        infinite  100    idle queue1-st-c5xlarge-[1-100]
```

Cuando un recurso de cómputo lo tenga `MinCount == MaxCount`, todos los nodos de cómputo correspondientes serán estáticos y todas las instancias se lanzarán en el momento de la creación o actualización del clúster y se mantendrán en funcionamiento. Por ejemplo:

```
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue1
      ComputeResources:
        - Name: c5xlarge
          Instances:
            - InstanceType: c5.xlarge
              MinCount: 100
              MaxCount: 100
```

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
```

```
queue1*    up    infinite    100    idle queue1-st-c5xlarge-[1-100]
```

Actualización de la capacidad del clúster

La actualización de la capacidad del clúster incluye añadir o eliminar colas, recursos informáticos o cambiar la capacidad MinCount/MaxCount de un recurso informático. A partir de la AWS ParallelCluster versión 3.9.0, para reducir el tamaño de una cola es necesario detener la flota informática o [QueueUpdateStrategy](#) configurarla en TERMINATE antes de que se lleve a cabo la actualización del clúster. No es necesario detener la flota de cómputo ni configurarla en TERMINATE [QueueUpdateStrategy](#) cuando:

- Añadir nuevas colas a Scheduling/ [SlurmQueues](#)
- Añadir nuevos recursos de cómputo a una cola Scheduling/ [SlurmQueues/ComputeResources](#)
- Aumentar la cantidad [MaxCount](#) de un recurso de cómputo
- Incremento MinCount de un recurso informático y aumento MaxCount del mismo recurso informático en al menos la misma cantidad

Consideraciones y limitaciones

El objetivo de esta sección es describir los factores, restricciones o limitaciones importantes que se deben tener en cuenta al redimensionar la capacidad del clúster.

- Al eliminar una cola de Scheduling/<https://docs.aws.amazon.com/parallelcluster/latest/ug/Scheduling-v3.html#Scheduling-v3-SlurmQueues> SlurmQueues todos los nodos de procesamiento con un nombre <Queue/Name> - *, tanto estático como dinámico, se eliminarán de la Slurm configuración y se eliminarán las instancias de EC2 correspondientes.
- Al eliminar un recurso informático Scheduling/SlurmQueues/<https://docs.aws.amazon.com/parallelcluster/latest/ug/Scheduling-v3.html#Scheduling-v3-SlurmQueues-ComputeResources> ComputeResources de una cola, todos los nodos informáticos con nombre <Queue/Name> - * - <ComputeResource/Name> - *, tanto estáticos como dinámicos, se eliminarán de la Slurm configuración y se eliminarán las instancias de EC2 correspondientes.

Al cambiar el `MinCount` parámetro de un recurso informático, podemos distinguir dos escenarios diferentes: si `MaxCount` se mantiene igual `MinCount` (solo capacidad estática) y si `MaxCount` es mayor que `MinCount` (capacidad estática y dinámica mixta).

La capacidad cambia solo con los nodos estáticos

- Si `MinCount == MaxCount`, al aumentar `MinCount` (y `MaxCount`), el clúster se configurará ampliando el número de nodos estáticos hasta el nuevo valor de `MinCount` `<Queue/Name>-st-<ComputeResource/Name>-<new_MinCount>` y el sistema seguirá intentando lanzar instancias EC2 para cumplir con la nueva capacidad estática requerida.
- Si `MinCount == MaxCount`, al disminuir `MinCount` (y `MaxCount`) la cantidad `N`, el clúster se configurará eliminando los últimos `N` nodos estáticos `<Queue/Name>-st-<ComputeResource/Name>-<old_MinCount - N>...<old_MinCount>`] y el sistema finalizará las instancias de EC2 correspondientes.

- Estado inicial `MinCount = MaxCount = 100`

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up       infinite   100    idle queue1-st-c5xlarge-[1-100]
```

- Actualización `-30` en `MinCount` y `MaxCount`: `MinCount = MaxCount = 70`

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up       infinite    70    idle queue1-st-c5xlarge-[1-70]
```

La capacidad cambia con nodos mixtos

Si `MinCount < MaxCount`, al aumentar `MinCount` en una cantidad `N` (suponiendo que `MaxCount` se mantenga sin cambios), el clúster se configurará ampliando el número de nodos estáticos hasta el nuevo valor de `MinCount` (`old_MinCount + N`): `<Queue/Name>-st-<ComputeResource/Name>-<old_MinCount + N>` y el sistema seguirá intentando lanzar instancias EC2 para cumplir con la nueva capacidad estática requerida. Además, para respetar la `MaxCount` capacidad

del recurso informático, la configuración del clúster se actualiza eliminando los últimos N nodos dinámicos, `<Queue/Name>-dy-<ComputeResource/Name>-[<MaxCount - old_MinCount - N>...<MaxCount - old_MinCount>]` y el sistema finalizará las instancias de EC2 correspondientes.

- Estado inicial: `MinCount = 100`; `MaxCount = 150`

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up    infinite   50    idle~ queue1-dy-c5xlarge-[1-50]
queue1*   up    infinite  100    idle queue1-st-c5xlarge-[1-100]
```

- Actualiza +30 a `MinCount` : `MinCount = 130` (`MaxCount = 150`)

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up    infinite   20    idle~ queue1-dy-c5xlarge-[1-20]
queue1*   up    infinite  130    idle queue1-st-c5xlarge-[1-130]
```

Si `MinCount < MaxCount`, al aumentar N `MinCount` y en `MaxCount` la misma cantidad, el clúster se configurará ampliando el número de nodos estáticos hasta el nuevo valor de `MinCount` (`old_MinCount + N`): `<Queue/Name>-st-<ComputeResource/Name>-<old_MinCount + N>` y el sistema seguirá intentando lanzar instancias EC2 para cumplir con la nueva capacidad estática requerida. Además, no se realizará ningún cambio en el número de nodos dinámicos para adaptarlo a los nuevos

Valor `MaxCount`.

- Estado inicial: `MinCount = 100`; `MaxCount = 150`

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up    infinite   50    idle~ queue1-dy-c5xlarge-[1-50]
```

```
queue1*      up    infinite    100   idle queue1-st-c5xlarge-[1-100]
```

- Actualiza +30 a MinCount : MinCount = 130 (MaxCount = 180)

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up     infinite   20    idle~ queue1-dy-c5xlarge-[1-50]
queue1*   up     infinite  130    idle queue1-st-c5xlarge-[1-130]
```

Si $\text{MinCount} < \text{MaxCount}$, al reducir la MinCount cantidad N (suponiendo que MaxCount se mantenga sin cambios), el clúster se configurará eliminando los últimos N nodos estáticos `<Queue/Name>-st-<ComputeResource/Name>-[<old_MinCount - N>...<old_MinCount>` y el sistema cerrará las instancias de EC2 correspondientes. Además, para respetar la MaxCount capacidad del recurso informático, la configuración del clúster se actualiza ampliando el número de nodos dinámicos para cubrir el vacío. $\text{MaxCount} - \text{new_MinCount}$: `<Queue/Name>-dy-<ComputeResource/Name>-[1..<MaxCount - new_MinCount>]` En este caso, dado que se trata de nodos dinámicos, no se lanzará ninguna nueva instancia de EC2 a menos que el planificador tenga tareas pendientes en los nuevos nodos.

- Estado inicial: MinCount = 100; MaxCount = 150

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up     infinite   50    idle~ queue1-dy-c5xlarge-[1-50]
queue1*   up     infinite  100    idle queue1-st-c5xlarge-[1-100]
```

- Actualización -30 el MinCount : MinCount = 70 (MaxCount = 120)

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up     infinite   80    idle~ queue1-dy-c5xlarge-[1-80]
```

```
queue1*      up    infinite    70    idle queue1-st-c5xlarge-[1-70]
```

Si $MinCount < MaxCount$, al disminuir $MinCount$ y con $MaxCount$ la misma cantidad N , el clúster se configurará eliminando los últimos N nodos estáticos `<Queue/Name>-st-<ComputeResource/Name>-<old_MinCount - N>...<oldMinCount>`] y el sistema finalizará las instancias de EC2 correspondientes.

Además, no se realizará ningún cambio en el número de nodos dinámicos para respetar el nuevo $MaxCount$ valor.

- Estado inicial: $MinCount = 100$; $MaxCount = 150$

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up      infinite   50    idle~ queue1-dy-c5xlarge-[1-50]
queue1*   up      infinite  100    idle queue1-st-c5xlarge-[1-100]
```

- Actualización -30 el $MinCount$: $MinCount = 70$ ($MaxCount = 120$)

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up      infinite   80    idle~ queue1-dy-c5xlarge-[1-50]
queue1*   up      infinite   70    idle queue1-st-c5xlarge-[1-70]
```

Si $MinCount < MaxCount$, al disminuir la $MaxCount$ cantidad N (suponiendo que $MinCount$ se mantenga sin cambios), el clúster se configurará eliminando los últimos N nodos dinámicos `<Queue/Name>-dy-<ComputeResource/Name>-<old_MaxCount - N>...<oldMaxCount>`] y el sistema cerrará las instancias de EC2 correspondientes en caso de que estuvieran en ejecución. No se espera ningún impacto en los nodos estáticos.

- Estado inicial: $MinCount = 100$; $MaxCount = 150$

- ```
$ sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
queue1* up infinite 50 idle~ queue1-dy-c5xlarge-[1-50]
queue1* up infinite 100 idle queue1-st-c5xlarge-[1-100]
```
- Actualización -30 el MaxCount : MinCount = 100 (MaxCount = 120)

- ```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up    infinite   20    idle~ queue1-dy-c5xlarge-[1-20]
queue1*   up    infinite  100    idle queue1-st-c5xlarge-[1-100]
```

Impactos en los puestos de trabajo

En todos los casos en los que se eliminen los nodos y se cierren las instancias de EC2, se volverá a poner en cola un trabajo por lotes que se ejecute en los nodos eliminados, a menos que no haya otros nodos que cumplan con los requisitos del trabajo. En este último caso, el trabajo fallará con el estado `NODE_FAIL` y desaparecerá de la cola; si es el caso, será necesario volver a enviarlo manualmente.

Si planea realizar una actualización de tamaño del clúster, puede impedir que los trabajos se ejecuten en los nodos que se van a eliminar durante la actualización planificada. Esto es posible configurando los nodos que se eliminarán durante el mantenimiento. Tenga en cuenta que poner un nodo en fase de mantenimiento no afectaría a los trabajos que eventualmente ya se estén ejecutando en el nodo.

Supongamos que con la actualización planificada para cambiar el tamaño del clúster [se va a eliminar el nodo `queueu-st-computeresource-[9-10]`]. Puede crear una Slurm reserva con el siguiente comando

```
sudo -i scontrol create reservation ReservationName=maint_for_update user=root
starttime=now duration=infinite flags=maint,ignore_jobs nodes=queueu-st-
computeresource-[9-10]
```

Esto creará una Slurm reserva con un nombre `maint_for_update` en los nodos `queue-st-computeresource-[9-10]`. Desde el momento en que se crea la reserva, no se pueden ejecutar más trabajos en los nodos `queue-st-computeresource-[9-10]`. Tenga en cuenta que la reserva no impedirá que los trabajos se asignen eventualmente a los nodos `queue-st-computeresource-[9-10]`.

Tras la actualización del tamaño del clúster, si la Slurm reserva se estableció solo en los nodos que se eliminaron durante la actualización del cambio de tamaño, la reserva de mantenimiento se eliminará automáticamente. Si, por el contrario, ha creado una Slurm reserva en los nodos que siguen presentes tras la actualización del cambio de tamaño del clúster, es posible que deseemos eliminar la reserva de mantenimiento de los nodos una vez realizada la actualización de tamaño mediante el siguiente comando

```
sudo -i scontrol delete ReservationName=maint_for_update
```

[Para obtener más información sobre la Slurm reserva, consulta el documento oficial de SchedMD aquí.](#)

Proceso de actualización del clúster sobre los cambios de capacidad

Tras un cambio en la configuración del programador, se ejecutan los siguientes pasos durante el proceso de actualización del clúster:

- Detenga AWS ParallelCluster `clustermgtd` (`supervisorctl stop clustermgtd`)
- Genere la configuración de Slurm particiones actualizada a partir de AWS ParallelCluster la configuración
- Reiniciar `slurmctld` (realizado mediante la receta del servicio Chef)
- Compruebe el `slurmctld` estado (`systemctl is-active --quiet slurmctld.service`)
- Recargar la configuración Slurm (`scontrol reconfigure`)
- Iniciar `clustermgtd` (`supervisorctl start clustermgtd`)

Para obtener información acerca de Slurm, consulte <https://slurm.schedmd.com>. Para obtener información sobre las descargas, consulte <https://github.com/SchedMD/slurm/tags>. Para ver el código fuente, consulte <https://github.com/SchedMD/slurm>.

AWS ParallelCluster versión (s)	Versión de Slurm compatible
3.9.0	23.11.4
3.8.0	23.02.7
3.7.2	23.02.6
3.7.1	23,02.5
3.7.0	23,02.4
3.6.0, 3.6.1	23,02.2
3.5.0, 3.5.1	22.05.8
3.4.0, 3.4.1	22.05.7
3.3.0, 3.3.1	22,05.5
3.1.4, 3.1.5, 3.2.0, 3.2.1	21,08,8-2
3.1.2, 3.1.3	21,08.6
3.1.1	21.08.5
3.0.0	20,11,8

Temas

- [Configuración de varias colas](#)
- [Guía de Slurm para el modo de cola múltiple](#)
- [Modo protegido de clúster Slurm](#)
- [Conmutación por error rápida de capacidad insuficiente en el clúster de Slurm](#)
- [Slurm programación basada en memoria](#)
- [Asignación de varios tipos de instancias con Slurm](#)
- [Escalado de clústeres para nodos dinámicos](#)
- [Slurmcontabilidad con AWS ParallelCluster](#)

- [Personalización de la configuración de Slurm](#)
- [Slurm prolog y epilog](#)
- [Tamaño y actualización de la capacidad del clúster](#)

Configuración de varias colas

Configuración de varias colas

Con AWS ParallelCluster versión 3, puede configurar varias colas estableciendo el [Scheduler](#) en `slurm` y especificando más de una cola para [SlurmQueues](#) en el archivo de configuración. En este modo, coexisten diferentes tipos de instancias en los nodos de computación que se especifican en la sección [ComputeResources](#) del archivo de configuración. [ComputeResources](#) con diferentes tipos de instancias se escalan de forma ascendente o descendente según sea necesario para [SlurmQueues](#).

Cuotas de recursos de computación y de colas de clúster

Resource	Cuota
Slurm queues	50 colas por clúster
Compute resources	50 recursos de computación por cola 50 recursos de computación por clúster

Recuentos de nodos

Cada recurso de computación de [ComputeResources](#) para una cola debe tener valores de [Name](#), [InstanceType](#), [MinCount](#) y [MaxCount](#) únicos. [MinCount](#) y [MaxCount](#) tienen valores predeterminados que definen el rango de instancias de un recurso de computación de [ComputeResources](#) para una cola. También puede especificar sus propios valores para [MinCount](#) y [MaxCount](#). Cada recurso de computación en [ComputeResources](#) está compuesto por nodos estáticos numerados del 1 al valor de [MinCount](#) y nodos dinámicos numerados del valor de [MinCount](#) al valor de [MaxCount](#).

Configuración de ejemplo

A continuación, se muestra un ejemplo de una sección de [programación](#) para un archivo de configuración de clúster. En esta configuración, hay dos colas denominadas `queue1` y `queue2` y cada una de ellas tiene un [ComputeResources](#) con un [MaxCount](#) especificado.

```
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue1
      ComputeResources:
        - InstanceType: c5.xlarge
          MaxCount: 5
          Name: c5xlarge
        - InstanceType: c4.xlarge
          MaxCount: 5
          Name: c4xlarge
    - Name: queue2
      ComputeResources:
        - InstanceType: c5.xlarge
          MaxCount: 5
          Name: c5xlarge
```

Nombres de hosts

Las instancias que se lanzan a la flota de computación se asignan de forma dinámica. Se generan nombres de host para cada nodo. De forma predeterminada, AWS ParallelCluster utilizará el siguiente formato de nombre de host:

```
$HOSTNAME=$QUEUE-$STATDYN-$COMPUTE_RESOURCE-$NODENUM
```

- `$QUEUE` es el nombre de la cola. Por ejemplo, si la sección [SlurmQueues](#) tiene una entrada con el valor [Name](#) establecido en “queue-name”, entonces “`$QUEUE`” es “queue-name”.
- `$STATDYN` es `st` para nodos estáticos o `dy` para dinámicos.
- `$COMPUTE_RESOURCE` es el [Name](#) del recurso de computación [ComputeResources](#) correspondiente a este nodo.
- `$NODENUM` es el número del nodo. `$NODENUM` está entre uno (1) y el valor de [MinCount](#) para los nodos estáticos y entre uno (1) y [MaxCount-MinCount](#) para los dinámicos.

En el ejemplo del archivo de configuración anterior, un nodo determinado de `queue1` y un recurso de computación `c5xlarge` tiene como nombre de host: `queue1-dy-c5xlarge-1`.

Tanto los nombres de host como los nombres de dominio completos (FQDN) se crean mediante las zonas alojadas de Amazon Route 53. El FQDN es `$HOSTNAME.$CLUSTERNAME.pcluster`, donde `$CLUSTERNAME` está el nombre del clúster.

Tenga en cuenta que también se utilizará el mismo formato para los nombres de los nodos de Slurm.

Los usuarios pueden optar por utilizar el nombre de host EC2 predeterminado de la instancia que alimenta el nodo de procesamiento en lugar del formato de nombre de host predeterminado que utiliza. AWS ParallelCluster Esto se puede hacer configurando el [UseEc2Hostnames](#) parámetro para que sea verdadero. Sin embargo, los nombres de los nodos de Slurm seguirán utilizando el formato predeterminado AWS ParallelCluster.

Guía de Slurm para el modo de cola múltiple

Aquí puede aprender cómo AWS ParallelCluster Slurm administrar los nodos de cola (partición) y cómo monitorear los estados de las colas y los nodos.

Información general

La arquitectura de escalado se basa en la [Guía de programación en la nube](#) de Slurm y en el complemento de ahorro de energía. Para obtener más información sobre el complemento de ahorro de energía, consulte la [Guía de ahorro de energía de Slurm](#). En la arquitectura, los recursos que podrían estar disponibles para un clúster suelen estar predefinidos en la configuración de Slurm como nodos de la nube.

Ciclo de vida de los nodos de la nube

A lo largo de su ciclo de vida, los nodos de la nube entran en varios de los siguientes estados (o en todos): `POWER_SAVING`, `POWER_UP` (`pow_up`), `ALLOCATED` (`alloc`) y `POWER_DOWN` (`pow_dn`). En algunos casos, un nodo de la nube puede entrar en el estado `OFFLINE`. La siguiente lista detalla varios aspectos de estos estados en el ciclo de vida de los nodos de la nube.

- Un nodo en un estado **POWER_SAVING** aparece con un sufijo `~` (por ejemplo `idle~`) en `sinfo`. En este estado, ninguna instancia de EC2 respalda al nodo. Sin embargo, Slurm sí puede asignar trabajos al nodo.
- Un nodo en transición a un estado **POWER_UP** aparece con un sufijo `#` (por ejemplo `idle#`) en `sinfo`. Un nodo pasa automáticamente a un estado `POWER_UP` cuando Slurm asigna un trabajo a un nodo en estado `POWER_SAVING`.

Como alternativa, puede pasar manualmente los nodos al estado `POWER_UP` como usuario raíz de su con el comando:

```
$ scontrol update nodename=nodename state=power_up
```

En esta etapa, se invoca a `ResumeProgram`, se lanzan y configuran las instancias de EC2 y el nodo pasa al estado `POWER_UP`.

- Un nodo que está actualmente disponible para su uso aparece sin sufijo (por ejemplo `idle`) en `sinfo`. Una vez que el nodo se haya configurado y se haya unido al clúster, estará disponible para ejecutar trabajos. En esta etapa, el nodo está correctamente configurado y listo para su uso.

Como regla general, se recomienda que el número de instancias de EC2 sea el mismo que el número de nodos disponibles. En la mayoría de los casos, los nodos estáticos están disponibles una vez creado el clúster.

- Un nodo que está en transición a un estado `POWER_DOWN` aparece con un sufijo `%` (por ejemplo `idle%`) en `sinfo`. Los nodos dinámicos entran automáticamente en el estado `POWER_DOWN` después de [ScaledownIdletime](#). Por el contrario, los nodos estáticos no están apagados en la mayoría de los casos. Sin embargo, como usuario raíz de `POWER_DOWN`, puede colocar los nodos en el estado su manualmente con el comando:

```
$ scontrol update nodename=nodename state=down reason="manual draining"
```

En este estado, se finalizan las instancias asociadas a un nodo, y el nodo vuelve a ese estado `POWER_SAVING` y está disponible para su uso después de [ScaledownIdletime](#).

El ajuste [ScaledownIdletime](#) se guarda en el ajuste `SuspendTimeout` de la configuración de Slurm.

- Un nodo que esté desconectado aparece con un sufijo `*` (por ejemplo `down*`) en `sinfo`. Un nodo se desconecta si el controlador de Slurm no puede contactar con el nodo o si los nodos estáticos están deshabilitados y las instancias de respaldo se finalizan.

Observe los estados de nodo que se muestran en el siguiente ejemplo de `sinfo`.

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
efa        up    infinite   4     idle~ efa-dy-efacompute1-[1-4]
```

```

efa          up    infinite    1   idle efa-st-efacompute1-1
gpu          up    infinite    1  idle% gpu-dy-gpucompute1-1
gpu          up    infinite    9  idle~ gpu-dy-gpucompute1-[2-10]
ondemand    up    infinite    2  mix#  ondemand-dy-ondemandcompute1-[1-2]
ondemand    up    infinite    18 idle~ ondemand-dy-ondemandcompute1-
[3-10],ondemand-dy-ondemandcompute2-[1-10]
spot*       up    infinite    13 idle~ spot-dy-spotcompute1-[1-10],spot-dy-
spotcompute2-[1-3]
spot*       up    infinite    2   idle spot-st-spotcompute2-[1-2]

```

Los nodos `spot-st-spotcompute2-[1-2]` y `efa-st-efacompute1-1` ya tienen instancias de respaldo configuradas y están disponibles para su uso. Los nodos `ondemand-dy-ondemandcompute1-[1-2]` se encuentran en el estado `POWER_UP` y deberían estar disponibles en unos minutos. El nodo `gpu-dy-gpucompute1-1` se encuentra en el estado `POWER_DOWN` y pasa al estado `POWER_SAVING` después de [ScaledownIdleTime](#) (el valor predeterminado es 10 minutos).

Todos los demás nodos se encuentran en el estado `POWER_SAVING` y no tienen instancias de EC2 que los respalden.

Trabajo con un nodo disponible

Un nodo disponible está respaldado por una instancia de EC2. De forma predeterminada, el nombre del nodo se puede usar para acceder directamente a la instancia mediante SSH (por ejemplo `ssh efa-st-efacompute1-1`). La dirección IP privada de la instancia se puede recuperar con el comando:

```
$ scontrol show nodes nodename
```

Compruebe la dirección IP en el campo `NodeAddr` devuelto.

En el caso de los nodos que no estén disponibles, el campo `NodeAddr` no debe apuntar a una instancia de EC2 en ejecución. Más bien, debe ser el mismo que el nombre del nodo.

Estados y envío de los trabajos

En la mayoría de los casos, los trabajos enviados se asignan inmediatamente a los nodos del sistema o se dejan pendientes si se asignan todos los nodos.

Si los nodos asignados a un trabajo incluyen algún nodo en un estado `POWER_SAVING`, el trabajo comienza con un estado `CF` o `CONFIGURING`. En este momento, el trabajo espera a que los nodos del estado `POWER_SAVING` pasen al estado `POWER_UP` y estén disponibles.

Una vez que todos los nodos asignados a un trabajo estén disponibles, el trabajo pasa al estado RUNNING (R).

De forma predeterminada, se envían todos los trabajos a la cola predeterminada (conocida como partición en Slurm). Esto se indica con un sufijo * después del nombre de la cola. Puede seleccionar una cola mediante la opción de envío de trabajos -p.

Todos los nodos están configurados con las siguientes características, que se pueden utilizar en los comandos de envío de trabajos:

- Un tipo de instancia (por ejemplo `c5.xlarge`)
- Un tipo de nodo (puede ser `dynamic` o `static`)

Puede ver las características de un nodo en particular mediante el comando:

```
$ scontrol show nodes nodename
```

En la devolución, consulte la lista `AvailableFeatures`.

Tenga en cuenta el estado inicial del clúster, que puede ver ejecutando el comando `sinfo`.

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
efa        up    infinite   4    idle~ efa-dy-efacompute1-[1-4]
efa        up    infinite   1    idle  efa-st-efacompute1-1
gpu        up    infinite  10    idle~ gpu-dy-gpucompute1-[1-10]
ondemand  up    infinite  20    idle~ ondemand-dy-ondemandcompute1-
[1-10],ondemand-dy-ondemandcompute2-[1-10]
spot*      up    infinite  13    idle~ spot-dy-spotcompute1-[1-10],spot-dy-
spotcompute2-[1-3]
spot*      up    infinite   2    idle  spot-st-spotcompute2-[1-2]
```

Tenga en cuenta que la cola predeterminada es `spot`. Se indica mediante el sufijo `*`.

Envíe un trabajo a un nodo estático de la cola predeterminada (`spot`).

```
$ sbatch --wrap "sleep 300" -N 1 -C static
```

Envíe un trabajo a un nodo dinámico de la cola EFA.

```
$ sbatch --wrap "sleep 300" -p efa -C dynamic
```

Envíe un trabajo a ocho (8) nodos c5.2xlarge y dos (2) nodos t2.xlarge de la cola ondemand.

```
$ sbatch --wrap "sleep 300" -p ondemand -N 10 -C "[c5.2xlarge*8&t2.xlarge*2]"
```

Envíe un trabajo a un nodo de GPU de la cola gpu.

```
$ sbatch --wrap "sleep 300" -p gpu -G 1
```

Tenga en cuenta el estado de los trabajos mediante el comando squeue.

```
$ squeue
```

JOBID	PARTITION	NAME	USER	ST	TIME	NODES	NODELIST(REASON)
12	ondemand	wrap	ubuntu	CF	0:36	10	ondemand-dy-ondemandcompute1-[1-8],ondemand-dy-ondemandcompute2-[1-2]
13	gpu	wrap	ubuntu	CF	0:05	1	gpu-dy-gpucompute1-1
7	spot	wrap	ubuntu	R	2:48	1	spot-st-spotcompute2-1
8	efa	wrap	ubuntu	R	0:39	1	efa-dy-efacompute1-1

Los trabajos 7 y 8 (en las colas spot y efa) ya se están ejecutando (R). Los trabajos 12 y 13 aún se están configurando (CF), probablemente esperando a que las instancias estén disponibles.

```
# Nodes states corresponds to state of running jobs
$ sinfo
```

PARTITION	AVAIL	TIMELIMIT	NODES	STATE	NODELIST
efa	up	infinite	3	idle~	efa-dy-efacompute1-[2-4]
efa	up	infinite	1	mix	efa-dy-efacompute1-1
efa	up	infinite	1	idle	efa-st-efacompute1-1
gpu	up	infinite	1	mix~	gpu-dy-gpucompute1-1
gpu	up	infinite	9	idle~	gpu-dy-gpucompute1-[2-10]
ondemand	up	infinite	10	mix#	ondemand-dy-ondemandcompute1-[1-8],ondemand-dy-ondemandcompute2-[1-2]
ondemand	up	infinite	10	idle~	ondemand-dy-ondemandcompute1-[9-10],ondemand-dy-ondemandcompute2-[3-10]
spot*	up	infinite	13	idle~	spot-dy-spotcompute1-[1-10],spot-dy-spotcompute2-[1-3]
spot*	up	infinite	1	mix	spot-st-spotcompute2-1
spot*	up	infinite	1	idle	spot-st-spotcompute2-2

Estado y características de los nodos

En la mayoría de los casos, los estados de los nodos se administran completamente de AWS ParallelCluster acuerdo con los procesos específicos del ciclo de vida de los nodos de la nube descritos anteriormente en este tema.

Sin embargo, AWS ParallelCluster también reemplaza o termina los nodos en mal estado y los DRAINED estados DOWN y nodos que tienen instancias de respaldo en mal estado. Para obtener más información, consulte [clustermgtd](#).

Estados de partición

AWS ParallelCluster admite los siguientes estados de partición. Una partición de Slurm es una cola en AWS ParallelCluster.

- UP: indica que la partición se encuentra en estado activo. Es el valor predeterminado de una partición. En este estado, todos los nodos de la partición están activos y disponibles para su uso.
- INACTIVE: indica que la partición se encuentra en estado inactivo. En este estado, se finalizan todas las instancias que respaldan a los nodos de una partición inactiva. No se lanzan nuevas instancias para los nodos de una partición inactiva.

clúster update-compute-fleet

- Detener la flota de procesamiento: cuando se ejecuta el siguiente comando, todas las particiones pasan al INACTIVE estado y AWS ParallelCluster los procesos mantienen las particiones en ese INACTIVE estado.

```
$ pcluster update-compute-fleet --cluster-name testSlurm \  
--region eu-west-1 --status STOP_REQUESTED
```

- Inicio de la flota de computación: cuando se ejecuta el siguiente comando, todas las particiones pasan inicialmente al estado UP. Sin embargo, AWS ParallelCluster los procesos no mantienen la partición en un UP estado. Debe cambiar los estados de las particiones manualmente. Todos los nodos estáticos están disponibles al cabo de unos minutos. Tenga en cuenta que establecer una partición en UP no activa ninguna capacidad dinámica.

```
$ pcluster update-compute-fleet --cluster-name testSlurm \  
--region eu-west-1 --status START_REQUESTED
```

Cuando se ejecuta `update-compute-fleet`, puede consultar el estado del clúster ejecutando el comando `pcluster describe-compute-fleet` y comprobando el `Status`. A continuación, se indican los estados posibles:

- `STOP_REQUESTED`: la solicitud de detener la flota de computación se envía al clúster.
- `STOPPING`: el proceso de `pcluster` detiene actualmente la flota de computación.
- `STOPPED`: el proceso de `pcluster` ha finalizado el proceso de detención, todas las particiones están en estado `INACTIVE` y todas las instancias de procesamiento han finalizado.
- `START_REQUESTED`: la solicitud de iniciar la flota de computación se envía al clúster.
- `STARTING`: el proceso de `pcluster` está iniciando actualmente el clúster.
- `RUNNING`: el proceso de `pcluster` ha finalizado el proceso de inicio, todas las particiones están en estado `UP` y los nodos estáticos están disponibles después de unos minutos.
- `PROTECTED`: este estado indica que algunas particiones tienen errores de arranque constantes. Las particiones afectadas están inactivas. Investigue el problema y, a continuación, ejecuta `update-compute-fleet` para volver a activar la flota.

Control manual de las colas

En algunos casos, es posible que quiera tener un control manual sobre los nodos o la cola (lo que se conoce como partición en Slurm) de un clúster. Puede administrar los nodos de un clúster mediante los siguientes procedimientos comunes usando el comando `scontrol`.

- Encienda los nodos dinámicos en estado **POWER_SAVING**

Ejecute el comando como usuario raíz de su:

```
$ scontrol update nodename=nodename state=power_up
```

También puede enviar un trabajo `sleep 1` de marcador de posición solicitando un número determinado de nodos y, a continuación, confíe en que Slurm activará el número de nodos requerido.

- Apague los nodos dinámicos antes de [ScaledownIdletime](#)

Se recomienda establecer los nodos dinámicos en `DOWN` como usuario raíz de su con el comando:

```
$ scontrol update nodename=nodename state=down reason="manually draining"
```

AWS ParallelCluster termina y restablece automáticamente los nodos dinámicos caídos.

En general, no es recomendable establecer nodos en `POWER_DOWN` directamente con el comando `scontrol update nodename=nodename state=power_down`. Esto se debe a que AWS ParallelCluster administra automáticamente el proceso de apagado.

- Deshabilite una cola (partición) o detenga todos los nodos estáticos de una partición específica

Establezca una cola específica en `INACTIVE` como usuario raíz de su con el comando:

```
$ scontrol update partition=queuename state=inactive
```

De este modo, se finalizan todas las instancias que respaldan a los nodos de la partición.

- Habilite una cola (partición)

Establezca una cola específica en `UP` como usuario raíz de su con el comando:

```
$ scontrol update partition=queuename state=up
```

Comportamiento y ajustes del escalado

A continuación, se muestra un ejemplo del flujo de trabajo del escalado normal:

- El programador recibe un trabajo que requiere dos nodos.
- El programador pasa dos nodos a un estado `POWER_UP` y llama a `ResumeProgram` con los nombres de los nodos (por ejemplo `queue1-dy-spotcompute1-[1-2]`).
- `ResumeProgram` lanza dos instancias de EC2 y asigna las direcciones IP privadas y los nombres de host de `queue1-dy-spotcompute1-[1-2]`, a la espera de `ResumeTimeout` (el periodo predeterminado es de 30 minutos antes de restablecer los nodos).
- Las instancias se configuran y se unen al clúster. Un trabajo comienza a ejecutarse en las instancias.
- El trabajo finaliza y deja de ejecutarse.
- Una vez transcurrido el `SuspendTime` configurado (que está establecido en [ScaledownIdleTime](#)), el programador establece las instancias en el estado `POWER_SAVING`. A continuación, el programador establece `queue1-dy-spotcompute1-[1-2]` en el estado `POWER_DOWN` y llama a `SuspendProgram` con los nombres de los nodos.

- Se llama a `SuspendProgram` para dos nodos. Los nodos permanecen en el estado `POWER_DOWN`, por ejemplo, permaneciendo `idle%` durante un `SuspendTimeout` (el periodo predeterminado es de 120 segundos, es decir, 2 minutos). Cuando `clustermgtd` detecta que los nodos se están apagando, finaliza las instancias de respaldo. Luego, pasa `queue1-dy-spotcompute1-[1-2]` al estado inactivo y restablece la dirección IP privada y el nombre de host para que estén listos para funcionar en futuros trabajos.

Si algo sale mal y no se puede lanzar una instancia para un nodo concreto por algún motivo, ocurre lo siguiente:

- El programador recibe un trabajo que requiere dos nodos.
- El programador pasa dos nodos de ampliación en la nube al estado `POWER_UP` y llama a `ResumeProgram` con los nombres de los nodos (por ejemplo `queue1-dy-spotcompute1-[1-2]`).
- `ResumeProgram` lanza solo una (1) instancia de EC2 y configura `queue1-dy-spotcompute1-1`, con una (1) instancia, `queue1-dy-spotcompute1-2`, que no se puede lanzarse.
- `queue1-dy-spotcompute1-1` no se ve afectado y entra en funcionamiento después de llegar al estado `POWER_UP`.
- `queue1-dy-spotcompute1-2` pasa al estado `POWER_DOWN` y el trabajo se vuelve a poner en cola automáticamente porque Slurm detecta un fallo en el nodo.
- `queue1-dy-spotcompute1-2` pasa a estar disponible después de `SuspendTimeout` (el valor predeterminado es 120 segundos, es decir, 2 minutos). Mientras tanto, el trabajo se vuelve a poner en cola y puede empezar a ejecutarse en otro nodo.
- El proceso anterior se repite hasta que el trabajo se pueda ejecutar en un nodo disponible sin que se produzca ningún error.

Hay dos parámetros de temporización que se pueden ajustar si es necesario:

- **ResumeTimeout** (el valor predeterminado es 30 minutos): `ResumeTimeout` controla el tiempo que Slurm espera antes de que el nodo pase al estado inactivo.
 - Podría ser útil ampliar el `ResumeTimeout` si el proceso previo o posterior a la instalación tiene una duración similar.
 - `ResumeTimeout` también es el tiempo máximo que AWS ParallelCluster espera antes de reemplazar o restablecer un nodo en caso de que haya algún problema. Los nodos de

cómputo se autofinalizan si se produce algún error durante el inicio o la configuración. AWS ParallelCluster los procesos sustituyen a un nodo al detectar una instancia terminada.

- **SuspendTimeout** (el valor predeterminado es 120 segundos, es decir, 2 minutos): SuspendTimeout controla la rapidez con la que los nodos se vuelven a colocar en el sistema y están listos para volver a usarse.
 - Un valor más bajo de SuspendTimeout significa que los nodos se restablecen más rápido y Slurm puede intentar lanzar instancias con más frecuencia.
 - Un valor más alto de SuspendTimeout significa que los nodos que han fallado se restablecen más lento. Mientras tanto, Slurm intenta usar otros nodos. Si SuspendTimeout es más de unos minutos, Slurm intenta recorrer todos los nodos del sistema. Un valor más alto de SuspendTimeout podría ser beneficioso para que los sistemas a gran escala (más de 1000 nodos) reduzcan el stress en Slurm cuando trata de volver a poner en cola los trabajos que fallan con frecuencia.
 - Ten en cuenta que SuspendTimeout esto no se refiere al tiempo de AWS ParallelCluster espera para finalizar una instancia de respaldo de un nodo. Las instancias de respaldo de los nodos POWER_DOWN finalizan inmediatamente. El proceso de finalización por lo general se completa en unos minutos. Sin embargo, durante este tiempo, el nodo permanece en el estado POWER_DOWN y no está disponible para que lo utilice el programador.

Registros para la arquitectura

La siguiente lista contiene los registros clave. El nombre del flujo de registro utilizado con Amazon CloudWatch Logs tiene el formato `{hostname}.{instance_id}.{logIdentifier}`, donde *LogIdentifier* sigue a los nombres de los registros.

- ResumeProgram: `/var/log/parallelcluster/slurm_resume.log` (slurm_resume)
- SuspendProgram: `/var/log/parallelcluster/slurm_suspend.log` (slurm_suspend)
- clustermgtd: `/var/log/parallelcluster/clustermgtd.log` (clustermgtd)
- computemgtd: `/var/log/parallelcluster/computemgtd.log` (computemgtd)
- slurmctld: `/var/log/slurmctld.log` (slurmctld)
- slurmd: `/var/log/slurmd.log` (slurmd)

Problemas frecuentes y cómo depurarlos:

Nodos que no se iniciaron o encendieron o que no se unieron al clúster

- **Nodos dinámicos:**
 - Compruebe el registro `ResumeProgram` para ver si se ha llamado a `ResumeProgram` con el nodo. Si no es así, compruebe el registro `slurmctld` para determinar si Slurm intentó llamar a `ResumeProgram` con el nodo. Tenga en cuenta que los permisos incorrectos activados en `ResumeProgram` pueden provocar un error silencioso.
 - Si se llama a `ResumeProgram`, compruebe si se ha lanzado una instancia para el nodo. Si la instancia no se ha lanzado, debería mostrarse un mensaje de error claro que explique por qué no se ha podido lanzar la instancia.
 - Si se ha lanzado una instancia, es posible que se haya producido algún problema durante el proceso de arranque. Busque la dirección IP privada y el ID de instancia correspondientes en el `ResumeProgram` registro y busque los registros de arranque correspondientes a la instancia específica en Logs. CloudWatch
- **Nodos estáticos:**
 - Compruebe el registro `clustermgtd` para ver si se han lanzado instancias para el nodo. Si no se han lanzado instancias, deberían mostrarse mensajes de error claros que expliquen por qué no se han podido lanzar las instancias.
 - Si se ha lanzado una instancia, se ha producido algún problema en el proceso de arranque. Busca la IP privada y el ID de instancia correspondientes en el `clustermgtd` registro y busca los registros de arranque correspondientes a la instancia específica en CloudWatch Logs.

Los nodos se han sustituido o finalizado de forma inesperada y han fallado

- **Nodos sustituidos o finalizados de forma inesperada:**
 - En la mayoría de los casos, `clustermgtd` administra todas las acciones de mantenimiento de los nodos. Para comprobar si `clustermgtd` ha sustituido o finalizado un nodo, compruebe el registro `clustermgtd`.
 - Si `clustermgtd` sustituye o finaliza el nodo, debería mostrarse un mensaje que indique el motivo de la acción. Si el motivo está relacionado con el programador (por ejemplo, el nodo estaba DOWN), consulte el registro `slurmctld` para obtener más información. Si el motivo está relacionado con EC2, utilice herramientas como Amazon CloudWatch o la consola, la CLI o los SDK de AWS EC2 para comprobar el estado o los registros de esa instancia. Por ejemplo, puede comprobar si la instancia tenía eventos programados o no pasó las comprobaciones de estado de EC2.
 - Si `clustermgtd` no finalizó el nodo, compruebe si `computemgtd` lo ha hecho o si EC2 ha finalizado la instancia para recuperar una instancia de spot.

- Fallos de nodo:

- En la mayoría de los casos, los trabajos se vuelven a poner en cola automáticamente si se produce un error en un nodo. Consulte el registro `slurmctld` para ver por qué ha fallado un trabajo o un nodo y evalúe la situación a partir de ahí.

Fallo al sustituir o finalizar instancias, error al apagar los nodos

- En general, `clustermgtd` administra todas las acciones de finalización de instancias esperadas. Consulte el registro `clustermgtd` para ver por qué no se ha podido sustituir o finalizar un nodo.
- En el caso de los nodos dinámicos que no superan el [ScaledownIdleTime](#), consulte el registro de `SuspendProgram` para ver si los procesos de `slurmctld` realizaron llamadas con el nodo específico como argumento. Tenga en cuenta que `SuspendProgram` no realiza ninguna acción específica. Más bien, solo se encarga de registrar cuando se le llama. Todas las finalizaciones y los restablecimientos de `NodeAddr` de las instancias los completa `clustermgtd`. Slurm pasa los nodos a IDLE después de `SuspendTimeout`.

Otros problemas:

- AWS ParallelCluster no toma decisiones de asignación de puestos ni de escalamiento. Solo intenta lanzar, finalizar y mantener los recursos de acuerdo con las instrucciones de Slurm.

Si tiene problemas relacionados con la asignación de trabajos, la asignación de nodos y la decisión de escalado, consulte el registro `slurmctld` para ver si hay errores.

Modo protegido de clúster Slurm

Cuando un clúster se ejecuta con el modo protegido activado, AWS ParallelCluster supervisa y rastrea los errores de arranque de los nodos de cómputo a medida que se lanzan los nodos de cómputo. Lo hace para detectar si estos errores se producen de forma continua.

Si se detecta lo siguiente en una cola (partición), el clúster pasa al estado protegido:

1. Los errores de arranque consecutivos de los nodos de cómputo se producen de forma continua y no se inicia correctamente el nodo de cómputo.
2. El recuento de errores alcanza un umbral predefinido.

Una vez que el clúster pasa al estado protegido, AWS ParallelCluster deshabilita las colas con errores iguales o superiores al umbral predefinido.

El modo protegido de clúster Slurm se agregó en AWS ParallelCluster versión 3.0.0.

Puede usar el modo protegido para reducir el tiempo y los recursos que se gastan en el ciclo de errores de arranque de los nodos de cómputo.

Parámetro de modo de modo protegido

protected_failure_count

`protected_failure_count` especifica el número de errores consecutivos en una cola (partición) que activan el estado de protección del clúster.

El valor predeterminado `protected_failure_count` es 10 y el modo protegido está activado.

Si `protected_failure_count` es mayor que cero, el modo protegido está activado.

Si `protected_failure_count` es inferior o igual a cero, el modo protegido está deshabilitado.

Puede cambiar el valor `protected_failure_count` añadiendo el parámetro en el archivo de configuración `clustermgtd` que se encuentra en `/etc/parallelcluster/slurm_plugin/parallelcluster_clustermgtd.conf` en el HeadNode.

Puede actualizar este parámetro en cualquier momento y no necesita detener la flota de cómputo para hacerlo. Si un lanzamiento se realiza correctamente en una cola antes de que llegue el recuento de errores `protected_failure_count`, el recuento de errores se restablece a cero.

Compruebe el estado del clúster en estado protegido

Cuando un clúster está en estado protegido, puede comprobar el estado de la flota informática y los estados de los nodos.

Calcule el estado de la flota

El estado de la flota informática se encuentra PROTECTED en un clúster que se ejecuta en estado protegido.

```
$ pcluster describe-compute-fleet --cluster-name <cluster-name> --region <region-id>
```

```
{
  "status": "PROTECTED",
  "lastStatusUpdateTime": "2022-04-22T00:31:24.000Z"
}
```

Estado del nodo

Para saber qué colas (particiones) tienen errores de arranque y tienen activado el estado de protección, inicie sesión en el clúster y ejecute el `sinfo` comando. Las particiones con errores de arranque iguales o superiores `protected_failure_count` están en ese estado. `INACTIVE` Las particiones sin errores de arranque iguales o superiores `protected_failure_count` se encuentran en ese UP estado y funcionan según lo previsto.

El estado de `PROTECTED` no afecta a los trabajos en ejecución. Si los trabajos se ejecutan en una partición con errores de arranque iguales o superiores `protected_failure_count`, la partición se establece en una `INACTIVE` vez finalizados los trabajos en ejecución.

Tenga en cuenta los estados de nodo que se muestran en el siguiente ejemplo.

```
$ sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
queue1* inact infinite 10 down% queue1-dy-c5xlarge-[1-10]
queue1* inact infinite 3490 idle~ queue1-dy-c5xlarge-[11-3500]
queue2 up infinite 10 idle~ queue2-dy-c5xlarge-[1-10]
```

La partición `queue1` es `INACTIVE` debido a que se detectaron 10 errores consecutivos de arranque de nodos de cómputo.

Las instancias situadas detrás de los nodos `queue1-dy-c5xlarge-[1-10]` se lanzaron pero no pudieron unirse al clúster debido a un estado incorrecto.

El clúster se encuentra en estado protegido.

Los errores de arranque `queue2` no afectan a la partición `queue1`. Está en el UP estado y aún puede ejecutar trabajos.

¿Cómo desactivar el estado de protección

Una vez resuelto el error de arranque, puede ejecutar el siguiente comando para que el clúster deje de estar protegido.

```
$ pcluster update-compute-fleet --cluster-name <cluster-name> \
  --region <region-id> \
  --status START_REQUESTED
```

Fallos de Bootstrap que activan el estado protegido

Los errores de Bootstrap que activan el estado protegido se subdividen en los tres tipos siguientes. Para identificar el tipo y el problema, puede comprobar si se AWS ParallelCluster generaron registros. Si se generaron registros, puedes comprobarlos para ver los detalles del error. Para obtener más información, consulte [Recuperación y conservación de registros](#).

1. Error de Bootstrap que provoca que una instancia se cierre automáticamente.

Una instancia falla al principio del proceso de arranque, como una instancia que se cierra automáticamente debido a errores en el script [SlurmQueues](#) \ [CustomActions](#) \ [OnNodeStart](#) | [OnNodeConfigured](#).

En el caso de los nodos dinámicos, busque errores similares a estos:

```
Node bootstrap error: Node ... is in power up state without valid backing instance
```

Para los nodos estáticos, busca errores similares a los siguientes en el `clustermgtd` log (`/var/log/parallelcluster/clustermgtd`):

```
Node bootstrap error: Node ... is in power up state without valid backing instance
```

2. El nodo `resume_timeout` o `node_replacement_timeout` caduca.

Una instancia no puede unirse al clúster dentro de `resume_timeout` (para nodos dinámicos) o `node_replacement_timeout` (para nodos estáticos). No se autotermina antes de que se agote el tiempo de espera. Por ejemplo, la red no está configurada correctamente para el clúster y el nodo se establece en ese DOWN estado una vez Slurm transcurrido el tiempo de espera.

En el caso de los nodos dinámicos, busque errores similares a estos:

```
Node bootstrap error: Resume timeout expires for node
```

Para los nodos estáticos, busca errores similares a los siguientes en el `clustermgtd` log (`/var/log/parallelcluster/clustermgtd`):

```
Node bootstrap error: Replacement timeout expires for node ... in replacement.
```

3. Los nodos no pasan la comprobación de estado.

Una instancia situada detrás del nodo no supera una comprobación de estado de EC2 o una comprobación de estado de un evento programado, y los nodos se consideran nodos con errores de arranque. En este caso, la instancia finaliza por un motivo ajeno al control de AWS ParallelCluster.

Busca errores similares a los siguientes en el `clustermgtd` registro (`/var/log/parallelcluster/clustermgtd`):

```
Node bootstrap error: Node %s failed during bootstrap when performing health check.
```

4. Los nodos de cómputo no se registran correctamente en Slurm.

El registro del daemon `slurmd` con el daemon de control Slurm (`slurmctld`) falla y hace que el estado del nodo de cómputo cambie al estado `INVALID_REG`. Los nodos de cómputo Slurm mal configurados pueden provocar este error, como los nodos computados configurados con errores de especificación de nodos de cómputo [CustomSlurmSettings](#).

Busque errores similares a los siguientes en el archivo de registro `slurmctld` (`/var/log/slurmctld.log`) del nodo principal o en el archivo de registro `slurmd` (`/var/log/slurmd.log`) del nodo de cómputo fallido:

```
Setting node %s to INVALID with reason: ...
```

Cómo depurar el modo protegido

Si el clúster está protegido y AWS ParallelCluster ha generado `clustermgtd` registros a partir de los nodos de procesamiento problemáticos `HeadNode` y los `cloud-init-output` registros de ellos, puede comprobar los detalles del error en los registros. Para obtener más información acerca de la recuperación de registros, consulte [Recuperación y conservación de registros](#).

Registro `clustermgtd` (`/var/log/parallelcluster/clustermgtd`) en el nodo principal

Los mensajes de registro muestran qué particiones tienen errores de arranque y el recuento de errores de arranque correspondiente.

```
[slurm_plugin.clustermgtd:_handle_protected_mode_process] - INFO - Partitions bootstrap failure count: {'queue1': 2}, cluster will be set into protected mode if protected failure count reach threshold.
```

En el registro `clustermgtd`, busque en `Found the following bootstrap failure nodes` qué nodo no se pudo iniciar.

```
[slurm_plugin.clustermgtd:_handle_protected_mode_process] - WARNING - Found the following bootstrap failure nodes: (x2) ['queue1-st-c5large-1(192.168.110.155)', 'broken-st-c5large-2(192.168.65.215)']
```

En el registro `clustermgtd`, busque en `Node bootstrap error` el motivo del error.

```
[slurm_plugin.clustermgtd:_is_node_bootstrap_failure] - WARNING - Node bootstrap error: Node broken-st-c5large-2(192.168.65.215) is currently in replacement and no backing instance
```

Registro **cloud-init-output** (`/var/log/cloud-init-output.log`) en los nodos de cómputo

Tras obtener la dirección IP privada del nodo de error de arranque en el registro `clustermgtd`, puede encontrar el registro del nodo de procesamiento correspondiente iniciando sesión en el nodo de procesamiento o siguiendo las instrucciones [Recuperación y conservación de registros](#) para recuperar los registros. En la mayoría de los casos, el registro `/var/log/cloud-init-output` del nodo problemático muestra el paso que provocó el error de arranque del nodo de cómputo.

Conmutación por error rápida de capacidad insuficiente en el clúster de Slurm

A partir de AWS ParallelCluster versión 3.2.0, los clústeres se ejecutan con el modo de conmutación por error rápida con capacidad insuficiente habilitado de forma predeterminada. Esto minimiza el tiempo que se tarda en volver a poner en cola un trabajo cuando se detectan errores de capacidad insuficiente de EC2. Esto resulta especialmente eficaz cuando se configura el clúster con varios tipos de instancias.

EC2 detectó fallos por capacidad insuficiente:

- `InsufficientInstanceCapacity`
- `InsufficientHostCapacity`
- `InsufficientReservedInstanceCapacity`

- `MaxSpotInstanceCountExceeded`
- `SpotMaxPriceTooLow`: se activa si el precio de solicitud de spot es menor que el precio mínimo requerido de cumplimiento de la solicitud de spot.
- `Unsupported`: se activa con el uso de un tipo de instancia que no sea compatible en una Región de AWS específica.

En el modo de conmutación rápida por error con capacidad insuficiente, si se detecta un error de capacidad insuficiente al asignar un trabajo a [SlurmQueues/compute_resource](#), AWS ParallelCluster hace lo siguiente:

1. Establece el recurso de computación en un estado deshabilitado (DOWN) durante un periodo de tiempo predefinido.
2. Utiliza `POWER_DOWN_FORCE` para cancelar los trabajos de los nodos fallidos del recurso de computación y para suspender el nodo fallido. Establece el nodo fallido en el estado IDLE y `POWER_DOWN (!)` y, después, en `POWERING_DOWN (%)`.
3. Vuelve a poner el trabajo en cola para enviarlo a otro recurso de computación.

Los nodos estáticos y encendidos del recurso de computación deshabilitado no se ven afectados. Los trabajos se pueden completar en estos nodos.

Este ciclo se repite hasta que el trabajo se asigne correctamente a uno o varios nodos de recursos de computación. Para obtener más información sobre los estados de los nodos, consulte [Guía de Slurm para el modo de cola múltiple](#).

Si no se encuentra ningún recurso de computación para ejecutar el trabajo, el trabajo se establece en el estado PENDING hasta que transcurra el periodo de tiempo predefinido. En este caso, puede modificar el periodo de tiempo predefinido tal y como se describe en la siguiente sección.

Parámetro de tiempo de espera de capacidad insuficiente

`insufficient_capacity_timeout`

`insufficient_capacity_timeout` especifica el periodo de tiempo (en segundos) durante el que el recurso de computación permanece en estado deshabilitado (down) cuando se detecta un error de capacidad insuficiente.

De forma predeterminada, `insufficient_capacity_timeout` está habilitado.

El valor predeterminado de `insufficient_capacity_timeout` es 600 segundos (10 minutos).

Si el valor de `insufficient_capacity_timeout` es menor o igual a cero, se deshabilita el modo de conmutación por error rápida con capacidad insuficiente.

Puede cambiar el valor de `insufficient_capacity_timeout` añadiendo el parámetro en el archivo de configuración `clustermgtd` ubicado en `/etc/parallelcluster/slurm_plugin/parallelcluster_clustermgtd.conf` en el `HeadNode`.

Se puede actualizar el parámetro en cualquier momento sin detener la flota de computación.

Por ejemplo:

- `insufficient_capacity_timeout=600`:

Si se detecta un error por capacidad insuficiente, el recurso de computación se establece como deshabilitado (DOWN). Transcurridos 10 minutos, el nodo fallido se establece en el estado `idle~` (POWER_SAVING).

- `insufficient_capacity_timeout=60`:

Si se detecta un error por capacidad insuficiente, el recurso de computación se establece como deshabilitado (DOWN). Transcurrido 1 minutos, su nodo fallido se establece en el estado `idle~`.

- `insufficient_capacity_timeout=0`:

El modo de conmutación por error rápida con capacidad insuficiente está deshabilitado. El recurso de computación no está deshabilitado.

Note

Es posible que haya un retraso de hasta un minuto entre el momento en que los nodos fallan debido a errores de capacidad insuficiente y el momento en que el daemon de administración de clústeres detecta los errores de los nodos. Esto se debe a que el daemon de administración de clústeres comprueba si hay errores de capacidad insuficiente en los nodos y establece el estado down de los recursos de computación en intervalos de un minuto.

Estado del modo de conmutación por error rápida de capacidad insuficiente

Cuando un clúster se encuentra en el modo de conmutación por error rápida con capacidad insuficiente, puede comprobar su estado y los estados de los nodos.

Estados de los nodos

Cuando se envía un trabajo a un nodo dinámico de recursos de computación y se detecta un error de capacidad insuficiente, el nodo se coloca en el estado `down#` con el motivo.

```
(Code:InsufficientInstanceCapacity)Failure when resuming nodes.
```

A continuación, los nodos apagados (nodos en estado `idle~`) se establecen en `down~` con el motivo.

```
(Code:InsufficientInstanceCapacity)Temporarily disabling node due to insufficient capacity.
```

El trabajo se vuelve a poner en cola en otros recursos de computación de la cola.

Los nodos estáticos del recurso de computación y los nodos que son UP no se ven afectados por el modo de conmutación por error rápida y con capacidad insuficiente.

Observe los estados de los nodos que se muestran en el siguiente ejemplo.

```
$ sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
queue1*   up    infinite   30  idle~ queue1-dy-c-1-[1-15],queue1-dy-c-2-[1-15]
queue2    up    infinite   30  idle~ queue2-dy-c-1-[1-15],queue2-dy-c-2-[1-15]
```

Enviamos un trabajo a `queue1` que requiere un nodo.

```
$ sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
queue1*   up    infinite   1   down# queue1-dy-c-1-1
queue1*   up    infinite  15  idle~ queue1-dy-c-2-[1-15]
queue1*   up    infinite  14  down~ queue1-dy-c-1-[2-15]
queue2    up    infinite  30  idle~ queue2-dy-c-1-[1-15],queue2-dy-c-2-[1-15]
```

Se lanza el nodo `queue1-dy-c-1-1` para ejecutar el trabajo. Sin embargo, no se pudo iniciar la instancia debido a un error de capacidad insuficiente. El nodo `queue1-dy-c-1-1` está configurado

en down. El nodo dinámico apagado dentro del recurso de computación (queue2-dy-c-1) está establecido en down.

Puede comprobar el motivo del nodo con `scontrol show nodes`.

```
$ scontrol show nodes queue1-dy-c-1-1
NodeName=broken-dy-c-2-1 Arch=x86_64 CoresPerSocket=1
CPUAlloc=0 CPUTot=96 CPULoad=0.00
...
ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s
Reason=(Code:InsufficientInstanceCapacity)Failure when resuming nodes
[root@2022-03-10T22:17:50]

$ scontrol show nodes queue1-dy-c-1-2
NodeName=broken-dy-c-2-1 Arch=x86_64 CoresPerSocket=1
CPUAlloc=0 CPUTot=96 CPULoad=0.00
...
ExtSensorsJoules=n/s ExtSensorsWatts=0 ExtSensorsTemp=n/s
Reason=(Code:InsufficientInstanceCapacity)Temporarily disabling node due to
insufficient capacity [root@2022-03-10T22:17:50]
```

El trabajo está en cola para otro tipo de instancias dentro de los recursos de computación de la cola.

Una vez transcurrido `insufficient_capacity_timeout`, los nodos del recurso de computación se restablecen a su estado `idle~`.

```
$ sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
queue1*   up    infinite    30  idle~ queue1-dy-c-1-[1-15],queue1-dy-c-2-[1-15]
queue2    up    infinite    30  idle~ queue2-dy-c-1-[1-15],queue2-dy-c-2-[1-15]
```

Una vez transcurrido `insufficient_capacity_timeout` y que los nodos del recurso de computación se hayan restablecido al estado `idle~`, el programador Slurm asigna a los nodos una prioridad inferior. El programador sigue seleccionando nodos de otros recursos de computación de cola con pesos más altos, a menos que se produzca una de las siguientes situaciones:

- Los requisitos de envío de un trabajo coinciden con el recurso de computación recuperado.
- No hay otros recursos de computación disponibles porque están al límite de su capacidad.
- `slurmctld` se reinicia.

- La flota de computación de AWS ParallelCluster se detiene y comienza a apagar y encender todos los nodos.

Registros relacionados

Los registros relacionados con los errores de capacidad insuficiente y el modo de conmutación rápida por error con capacidad insuficiente se encuentran en el registro `resume` y el registro `clustermgtd` de Slurm del nodo principal.

Slurm `resume` (`/var/log/parallelcluster/slurm_resume.log`)

Mensajes de error cuando un nodo no se inicia debido a una capacidad insuficiente.

```
[slurm_plugin.instance_manager:_launch_ec2_instances] - ERROR - Failed RunInstances request: dcd0c252-90d4-44a7-9c79-ef740f7ecd87
[slurm_plugin.instance_manager:add_instances_for_nodes] - ERROR - Encountered exception when launching instances for nodes (x1) ['queue1-dy-c-1-1']: An error occurred
(InsufficientInstanceCapacity) when calling the RunInstances operation (reached max retries: 1): We currently do not have sufficient p4d.24xlarge capacity in the Availability Zone you requested (us-west-2b). Our system will be working on provisioning additional capacity. You can currently get p4d.24xlarge capacity by not specifying an Availability Zone in your request or choosing us-west-2a, us-west-2c.
```

Slurm `clustermgtd` (`/var/log/parallelcluster/clustermgtd`)

El recurso de computación c-1 de la `queue1` está deshabilitado debido a una capacidad insuficiente.

```
[slurm_plugin.clustermgtd:_reset_timeout_expired_compute_resources] - INFO - The following compute resources are in down state due to insufficient capacity: {'queue1': {'c-1': ComputeResourceFailureEvent(timestamp=datetime.datetime(2022, 4, 14, 23, 0, 4, 769380), tzinfo=datetime.timezone.utc), error_code='InsufficientInstanceCapacity')}}}, compute resources are reset after insufficient capacity timeout (600 seconds) expired
```

Una vez transcurrido el tiempo de espera de capacidad insuficiente, el recurso de computación se restablece y los nodos de los recursos de computación se establecen en `idle~`.

```
[root:_reset_insufficient_capacity_timeout_expired_nodes] - INFO - Reset the
following compute resources because insufficient capacity
timeout expired: {'queue1': ['c-1']}
```

Slurm programación basada en memoria

A partir de la versión 3.2.0, AWS ParallelCluster admite la programación Slurm basada en memoria con el parámetro de configuración [SlurmSettings/EnableMemoryBasedScheduling](#)cluster.

Note

[A partir de AWS ParallelCluster la versión 3.7.0, se EnableMemoryBasedScheduling puede habilitar si configuras varios tipos de instancias en Instances.](#)

Para AWS ParallelCluster las versiones 3.2.0 a 3.6. x, no se EnableMemoryBasedScheduling puede habilitar si configuras varios tipos de [instancias en Instances](#).

Warning

Cuando especificas varios tipos de instancias en un recurso de cómputo de Slurm cola con la EnableMemoryBasedScheduling opción habilitada, el RealMemory valor es la cantidad mínima de memoria disponible para todos los tipos de instancias. Si especificas tipos de instancias con capacidades de memoria muy diferentes, esto podría provocar que se desperdicie una cantidad significativa de memoria.

Con EnableMemoryBasedScheduling: true, el programador Slurm realiza un seguimiento de la cantidad de memoria que requiere cada trabajo en cada nodo. A continuación, el programador de Slurm utiliza esta información para programar varios trabajos en el mismo nodo de computación. La cantidad total de memoria que requieren los trabajos en un nodo no puede ser mayor que la memoria de nodo disponible. El programador evita que un trabajo utilice más memoria de la que se solicitó cuando se envió el trabajo.

Con EnableMemoryBasedScheduling: false, los trabajos podrían competir por la memoria en un nodo compartido y provocar errores y out-of-memory eventos en los trabajos.

⚠ Warning

Slurm utiliza una notación de potencia de 2 para sus etiquetas, como MB o GB. Lea estas etiquetas como MiB y GiB, respectivamente.

Slurm programación basada en la configuración y en la memoria

Con `EnableMemoryBasedScheduling: true`, Slurm establece los siguientes parámetros de Slurm configuración:

- [SelectTypeParameters=CR_CPU_Memory](#) en la `slurm.conf`. Esta opción configura la memoria del nodo para que sea un recurso consumible Slurm.
- [ConstrainRAMSpace=yes](#) en `Slurm.cgroup.conf`. Con esta opción, el acceso de un trabajo a la memoria se limita a la cantidad de memoria que el trabajo solicitó al enviarse.

ℹ Note

Otros parámetros Slurm de configuración pueden afectar al comportamiento del Slurm programador y del administrador de recursos cuando se configuran estas dos opciones. Para obtener más información, consulte la [documentación de Slurm](#).

Slurm programador y programación basada en memoria

EnableMemoryBasedScheduling: false (predeterminado)

De forma predeterminada, `EnableMemoryBasedScheduling` se establece en "false". Si es falsa, Slurm no incluye la memoria como recurso en su algoritmo de programación y no rastrea la memoria que utilizan los trabajos. Los usuarios pueden especificar la `--mem MEM_PER_NODE` opción de establecer la cantidad mínima de memoria por nodo que requiere un trabajo. Esto obliga al programador a elegir nodos con un `RealMemory` valor de al menos `MEM_PER_NODE` al programar el trabajo.

Por ejemplo, supongamos que un usuario envía dos trabajos con `--mem=5GB`. Si los recursos solicitados, como CPU o GPU, están disponibles, los trabajos se pueden ejecutar al mismo tiempo en un nodo con 8 GiB de memoria. Los dos trabajos no están programados en nodos de procesamiento con menos de 5 GiB de `RealMemory`.

⚠ Warning

Cuando la programación basada en memoria está deshabilitada, Slurm no registra la cantidad de memoria que utilizan los trabajos. Los trabajos que se ejecutan en el mismo nodo pueden competir por los recursos de memoria y provocar que el otro trabajo falle. Cuando la programación basada en memoria está deshabilitada, se recomienda a los usuarios que no especifiquen las opciones `--mem-per-cpu` o `--mem-per-gpu`. Estas opciones pueden provocar un comportamiento diferente al descrito en la [documentación de Slurm](#).

EnableMemoryBasedScheduling: true

Cuando `EnableMemoryBasedScheduling` se establece en “true”, Slurm realiza un seguimiento del uso de memoria de cada trabajo e impide que los trabajos consuman más memoria de la solicitada con las opciones de `--mem` envío.

En el ejemplo anterior, un usuario envía dos trabajos con `--mem=5GB`. Los trabajos no se pueden ejecutar al mismo tiempo en un nodo con 8 GiB de memoria. Esto se debe a que la cantidad total de memoria necesaria es mayor que la memoria disponible en el nodo.

Con la programación basada en memoria habilitada `--mem-per-cpu` y `--mem-per-gpu` compórtese de forma coherente con lo que se describe en la documentación de Slurm. Por ejemplo, un trabajo se envía con `--ntasks-per-node=2 -c 1 --mem-per-cpu=2GB`. En este caso, Slurm asigna al trabajo un total de 4 GiB para cada nodo.

⚠ Warning

Cuando la programación basada en memoria está habilitada, recomendamos que los usuarios incluyan una especificación `--mem` al enviar un trabajo. Con la Slurm configuración predeterminada incluida AWS ParallelCluster, si no se incluye ninguna opción de memoria (`--mem`, o `--mem-per-gpu`) `--mem-per-cpu`, Slurm asigna toda la memoria de los nodos asignados a la tarea, incluso si solo solicita una parte de los demás recursos, como las CPU o las GPU. De esta forma, se evita que se compartan nodos hasta que finalice el trabajo, ya que no hay memoria disponible para otros trabajos. Esto sucede porque Slurm establece la memoria por nodo para el trabajo a `DefMemPerNode` en un momento en que no se proporcionan especificaciones de memoria en el momento de enviar el trabajo. El valor

predeterminado de este parámetro es 0 y especifica un acceso ilimitado a la memoria de un nodo.

Si hay varios tipos de recursos informáticos con diferentes cantidades de memoria disponibles en la misma cola, es posible que a un trabajo enviado sin opciones de memoria se le asignen diferentes cantidades de memoria en distintos nodos. Esto depende de los nodos que el programador ponga a disposición del trabajo. Los usuarios pueden definir un valor personalizado para las opciones, como `DefMemPerNode` o [DefMemPerCPU](#), a nivel de clúster o partición, en los archivos de configuración Slurm para evitar este comportamiento.

Slurm RealMemory y AWS ParallelCluster SchedulableMemory

Con la Slurm configuración incluida AWS ParallelCluster, se Slurm interpreta como [RealMemory](#) la cantidad de memoria por nodo disponible para las tareas. [A partir de la versión 3.2.0, de forma predeterminada, se AWS ParallelCluster establece RealMemory en el 95 por ciento de la memoria que aparece en los tipos de instancias de Amazon EC2 y que devuelven los tipos de API de Amazon EC2. DescribeInstance](#)

Cuando la programación basada en memoria está deshabilitada, el Slurm programador se utiliza `RealMemory` para filtrar los nodos cuando los usuarios envían un trabajo según lo especificado. --mem

Cuando la programación basada en memoria está habilitada, el Slurm programador interpreta que es la cantidad máxima de memoria disponible `RealMemory` para los trabajos que se ejecutan en el nodo de cómputo.

Es posible que la configuración predeterminada no sea óptima para todos los tipos de instancias:

- Esta configuración puede ser superior a la cantidad de memoria a la que realmente pueden acceder los nodos. Esto puede suceder cuando los nodos de cómputo son tipos de instancias pequeñas.
- Esta configuración puede ser superior a la cantidad de memoria a la que realmente pueden acceder los nodos. Esto puede ocurrir cuando los nodos de cómputo son tipos de instancias grandes y puede generar una cantidad significativa de memoria no utilizada.

Puede usar [SlurmQueues/ComputeResources/SchedulableMemory](#) para ajustar el valor de `RealMemory` configure by para los nodos de cómputo. AWS ParallelCluster Para anular el valor

predeterminado, defina un valor personalizado específico para `SchedulableMemory` para la configuración de su clúster.

Para comprobar la memoria disponible real de un nodo de cómputo, ejecuta el `/opt/slurm/sbin/slurmd -C` comando en el nodo. Este comando devuelve la configuración de hardware del nodo, incluido el [RealMemory](#) valor. Para obtener más información, consulte [slurmd -C](#).

Asegúrese de que los procesos del sistema operativo del nodo de cómputo tengan suficiente memoria. Para ello, limite la memoria disponible para los trabajos configurando el `SchedulableMemory` valor en un valor inferior al `RealMemory` valor devuelto por el comando `slurmd -C`.

Asignación de varios tipos de instancias con Slurm

A partir de la versión 3.3.0 de AWS ParallelCluster, puede configurar el clúster para que asigne a partir del conjunto de tipos de instancias definidos de un recurso de computación. La asignación puede basarse en estrategias de bajo coste o capacidad óptima para la flota de EC2.

Este conjunto de tipos de instancias definidos debe tener el mismo número de vCPU o, si el subprocesamiento múltiple está deshabilitado, el mismo número de núcleos. Además, este conjunto de tipos de instancias debe tener el mismo número de aceleradores de los mismos fabricantes. Si [Efa/Enabled](#) se establece en `true`, las instancias deben ser compatibles con EFA. Para obtener más información y conocer los requisitos, consulte [Scheduling/SlurmQueues/AllocationStrategy](#) y [ComputeResources/Instances](#).

Puede establecer [AllocationStrategy](#) en `lowest-price` o `capacity-optimized` función de su configuración de [CapacityType](#).

En [Instances](#), puede configurar un conjunto de tipos de instancias.

Note

A partir de la versión 3.7.0 de AWS ParallelCluster, se puede habilitar `EnableMemoryBasedScheduling` si configura varios tipos de instancias en [Instancias](#). En el caso de las versiones de la 3.2.0 a la 3.6.x de AWS ParallelCluster, no se puede habilitar `EnableMemoryBasedScheduling` si configura varios tipos de instancias en [Instancias](#).

En los siguientes ejemplos se muestra cómo consultar los tipos de instancias de las vCPU, la compatibilidad con EFA y la arquitectura.

Consulte InstanceTypes con 96 vCPU y arquitectura x86_64.

```
$ aws ec2 describe-instance-types --region region-id \
  --filters "Name=vcpu-info.default-vcpus,Values=96" "Name=processor-info.supported-
architecture,Values=x86_64" \
  --query "sort_by(InstanceTypes[*].
{InstanceType:InstanceType,MemoryMiB:MemoryInfo.SizeInMiB,CurrentGeneration:CurrentGeneration,V
&InstanceType})" \
  --output table
```

Consulte InstanceTypes con 64 núcleos, compatibilidad con EFA y arquitectura arm64.

```
$ aws ec2 describe-instance-types --region region-id \
  --filters "Name=vcpu-info.default-cores,Values=64" "Name=processor-
info.supported-architecture,Values=arm64" "Name=network-info.efa-
supported,Values=true" --query "sort_by(InstanceTypes[*].
{InstanceType:InstanceType,MemoryMiB:MemoryInfo.SizeInMiB,CurrentGeneration:CurrentGeneration,V
&InstanceType})" \
  --output table
```

En el siguiente ejemplo de fragmento de configuración de clústeres se muestra cómo puede utilizar estas propiedades InstanceType y AllocationStrategy.

```
...
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue-1
      CapacityType: ONDEMAND
      AllocationStrategy: lowest-price
      ...
  ComputeResources:
    - Name: computeresource1
      Instances:
        - InstanceType: r6g.2xlarge
        - InstanceType: m6g.2xlarge
        - InstanceType: c6g.2xlarge
      MinCount: 0
      MaxCount: 500
```

```
- Name: computeresource2
  Instances:
    - InstanceType: m6g.12xlarge
    - InstanceType: x2gd.12xlarge
  MinCount: 0
  MaxCount: 500
...
```

Escalado de clústeres para nodos dinámicos

ParallelCluster admite Slurm los métodos para escalar los clústeres de forma dinámica mediante Slurm el complemento de ahorro de energía. Para obtener más información, consulte la [Guía de programación en la nube](#) y la [Guía de ahorro de energía de Slurm](#) en la documentación de Slurm.

A partir de la ParallelCluster versión 3.8.0, ParallelCluster utiliza la reanudación a nivel de trabajo o el escalado a nivel de trabajo como estrategia de asignación dinámica de nodos predeterminada para escalar el clúster: ParallelCluster amplía el clúster en función de los requisitos de cada trabajo, la cantidad de nodos asignados al trabajo y los nodos que deben reanudarse. ParallelCluster obtiene esta información de la variable de entorno `SLURM_RESUME_FILE`.

El escalado de los nodos dinámicos es un proceso de dos pasos, que implica el lanzamiento de las instancias EC2 y la asignación de las instancias EC2 lanzadas a los nodos de Slurm. Cada uno de estos dos pasos se puede realizar mediante una all-or-nothing lógica o con el máximo esfuerzo.

Para el lanzamiento de las instancias EC2:

- all-or-nothing llama a la API EC2 de lanzamiento con un objetivo mínimo igual a la capacidad total del objetivo
- best-effort llama a la API de EC2 de lanzamiento con un objetivo mínimo igual a 1 y la capacidad objetivo total igual a la capacidad solicitada

Para asignar las instancias EC2 a los nodos de Slurm:

- all-or-nothing asigna instancias EC2 a los nodos de Slurm solo si es posible asignar una instancia de EC2 a cada nodo solicitado
- best-effort asigna instancias de EC2 a los nodos de Slurm incluso si todos los nodos solicitados no están cubiertos por la capacidad de la instancia de EC2

Las posibles combinaciones de las estrategias anteriores se traducen en las estrategias de lanzamiento. ParallelCluster

Example

<caption>The available ParallelCluster estrategias de lanzamiento that can be set into the [ScalingStrategy](#) cluster configuration to be used with escalamiento a nivel de puesto are:</caption>

all-or-nothingescalado:

Esta estrategia implica AWS ParallelCluster iniciar una llamada a la API de la instancia de lanzamiento de Amazon EC2 para cada trabajo, que requiere todas las instancias necesarias para que los nodos de cómputo solicitados se lancen correctamente. Esto garantiza que el clúster se escale solo cuando esté disponible la capacidad requerida por trabajo, lo que evita que queden instancias inactivas al final del proceso de escalado.

La estrategia utiliza una all-or-nothinglógica para el lanzamiento de las instancias de EC2 para cada trabajo y una all-or-nothinglógica para la asignación de las instancias de EC2 a los nodos de Slurm.

La estrategia agrupa las solicitudes de lanzamiento en lotes, uno por cada recurso informático solicitado y hasta 500 nodos cada uno. Para las solicitudes que abarcan varios recursos informáticos o que superan los 500 nodos, procesa varios lotes de ParallelCluster forma secuencial.

Si se produce un error en un lote de un solo recurso, se interrumpe toda la capacidad no utilizada asociada, lo que garantiza que no quede ninguna instancia inactiva al final del proceso de escalado.

Limitaciones

- El tiempo necesario para escalar es directamente proporcional al número de trabajos enviados por ejecución del programa de currículum de Slurm.
- La operación de escalado está limitada por el límite de la cuenta de RunInstances recursos, establecido en 1000 instancias de forma predeterminada. Esta limitación está de acuerdo con las políticas de regulación AWS de las API de EC2. Para obtener más información, consulte la documentación de regulación de las API de [AWS EC2](#)
- Cuando envía un trabajo en un recurso informático con un solo tipo de instancia, en una cola que abarca varias zonas de disponibilidad, la llamada a la API de lanzamiento de all-or-nothingEC2 solo se realiza correctamente si se puede proporcionar toda la capacidad en una única zona de disponibilidad.
- Al enviar un trabajo en un recurso informático con varios tipos de instancias, en una cola con una única zona de disponibilidad, la llamada a la API de lanzamiento de all-or-nothingEC2 solo se realiza correctamente si un único tipo de instancia puede proporcionar toda la capacidad.

- Cuando envías un trabajo en un recurso informático con varios tipos de instancias, en una cola que abarca varias zonas de disponibilidad, no se admite la llamada a la API de lanzamiento de all-or-nothingEC2 y, en cambio, se realiza el escalado al máximo. ParallelCluster

greedy-all-or-nothingescalado:

Esta variante de la all-or-nothing estrategia sigue garantizando que el clúster escale solo cuando esté disponible la capacidad requerida por trabajo, lo que evita las instancias inactivas al final del proceso de escalado, pero implica ParallelCluster iniciar una llamada a la API de la instancia de lanzamiento de Amazon EC2 que apunta a una capacidad objetivo mínima de 1, intentando maximizar la cantidad de nodos lanzados hasta la capacidad solicitada. La estrategia utiliza una lógica de máximo esfuerzo para el lanzamiento de las instancias EC2 para todos los trabajos, además de la all-or-nothinglógica para la asignación de las instancias EC2 a los nodos de Slurm para cada trabajo.

La estrategia agrupa las solicitudes de lanzamiento en lotes, uno por cada recurso informático solicitado y hasta 500 nodos cada uno. Para las solicitudes que abarcan varios recursos informáticos o que superan los 500 nodos, Parallelcluster procesa varios lotes de forma secuencial.

Garantiza que no quede ninguna instancia inactiva al final del proceso de escalado, maximizando el rendimiento a costa de una sobreescalabilidad temporal durante el proceso de escalado.

Limitaciones

- Es posible que se produzca un sobreescalado temporal, lo que conllevará costes adicionales para las instancias que pasen a un estado de ejecución antes de completar el escalado.
- Se aplica el mismo límite de instancias que en la all-or-nothing estrategia, sujeto al límite AWS de la cuenta de RunInstances recursos.

escalado con el máximo esfuerzo:

Esta estrategia se denomina llamada a la API de instancia de lanzamiento de EC2 y se orienta a una capacidad mínima de 1 persona y tiene como objetivo alcanzar la capacidad total solicitada a costa de dejar las instancias inactivas tras la ejecución del proceso de escalado si no está disponible toda la capacidad solicitada. La estrategia utiliza una lógica de máximo esfuerzo para lanzar las instancias EC2 para todos los trabajos, además de la lógica de máximo esfuerzo para asignar las instancias de Amazon EC2 a los nodos de Slurm para cada trabajo.

La estrategia agrupa las solicitudes de lanzamiento en lotes, uno por cada recurso informático solicitado y hasta 500 nodos cada uno. Para las solicitudes que abarcan varios recursos informáticos o que superan los 500 nodos, procesa varios lotes de ParallelCluster forma secuencial.

Esta estrategia permite escalar con creces el límite predeterminado de 1000 instancias en múltiples ejecuciones de procesos de escalado, a costa de tener instancias inactivas en los diferentes procesos de escalado.

Limitaciones

- Es posible que las instancias estén inactivas al final del proceso de escalado, en el caso de que no sea posible asignar todos los nodos solicitados por los trabajos.

El siguiente es un ejemplo que muestra cómo se comporta el escalado de los nodos dinámicos utilizando las diferentes estrategias de ParallelCluster lanzamiento. Supongamos que ha enviado dos trabajos solicitando 20 nodos cada uno, es decir, un total de 40 nodos del mismo tipo, pero solo hay 30 instancias de EC2 disponibles para cubrir la capacidad solicitada en EC2.

all-or-nothingescalado:

- Para el primer trabajo, se llama a una API de instancia de lanzamiento de all-or-nothingEC2 y se solicitan 20 instancias. Una llamada exitosa tiene como resultado el lanzamiento de 20 instancias
- all-or-nothing La asignación de las 20 instancias lanzadas a los nodos de Slurm para el primer trabajo se realizó correctamente
- Se llama a otra API de instancias de lanzamiento de all-or-nothingEC2, que solicita 20 instancias para el segundo trabajo. La llamada no se ha realizado correctamente, ya que solo hay capacidad para otras 10 instancias. No se ha lanzado ninguna instancia en este momento

greedy-all-or-nothingescalado:

- Se utiliza la API de instancia de lanzamiento de EC2 más adecuada, que solicita 40 instancias, que es la capacidad total solicitada por todos los trabajos. Esto da como resultado el lanzamiento de 30 instancias
- La all-or-nothing asignación de 20 de las instancias lanzadas a los nodos de Slurm para el primer trabajo se realiza correctamente

- Se intenta volver a all-or-nothing asignar las instancias lanzadas restantes a los nodos de Slurm para el segundo trabajo, pero dado que solo hay 10 instancias disponibles del total de 20 solicitadas por el trabajo, la asignación no se realiza correctamente
- Las 10 instancias lanzadas no asignadas finalizan

escalado al máximo:

- Se denomina API de instancia de lanzamiento de EC2, que solicita 40 instancias, que es la capacidad total solicitada por todos los trabajos. Esto da como resultado el lanzamiento de 30 instancias.
- Lo mejor es asignar 20 de las instancias lanzadas a los nodos de Slurm para el primer trabajo.
- Otro de los mejores esfuerzos consiste en asignar las 10 instancias lanzadas restantes a los nodos de Slurm para el segundo trabajo, que se ha realizado correctamente, incluso si la capacidad total solicitada era de 20. Sin embargo, dado que el trabajo solicitaba los 20 nodos y solo era posible asignar instancias de EC2 a 10 de ellos, el trabajo no puede iniciarse y las instancias se quedan inactivas hasta que se encuentre suficiente capacidad para iniciar las 10 instancias que faltan en una llamada posterior del proceso de escalado, o el programador programe el trabajo en otros nodos de cómputo que ya estén en ejecución.

Slurm estrategias de asignación dinámica de nodos en la versión 3.7.x

ParallelCluster utiliza dos tipos de estrategias de asignación dinámica de nodos para escalar el clúster:

- Asignación basada en la información de nodos solicitada disponible:
 - Reanudación de todos los nodos o escalado de lista de nodos:

ParallelCluster amplía el clúster basándose únicamente en los nombres Slurm de la lista de nodos solicitada cuando Slurm ResumeProgram se ejecuta. Asigna recursos de computación a los nodos solo por nombre de nodo. La lista de nombres de nodos puede abarcar varios trabajos.

- Reanudación a nivel de trabajo o escalado a nivel de trabajo:

ParallelCluster amplía el clúster en función de los requisitos de cada trabajo, el número actual de nodos que están asignados al trabajo y los nodos que deben reanudarse. ParallelCluster obtiene esta información de la variable de SLURM_RESUME_FILE entorno.

- Asignación con una estrategia de lanzamiento de EC2:

- Escalado óptimo:

ParallelCluster amplía el clúster mediante una llamada a la API de la instancia de lanzamiento de EC2 con una capacidad de destino mínima igual a 1, para lanzar algunas instancias, pero no necesariamente todas, necesarias para admitir los nodos solicitados.

- Un ll-or-nothing escalamiento:

ParallelCluster amplía el clúster mediante una llamada a la API de la instancia de lanzamiento de EC2 que solo tiene éxito si se lanzan todas las instancias necesarias para admitir los nodos solicitados. En este caso, llama a la API de la instancia de lanzamiento de EC2 con una capacidad de destino mínima igual a la capacidad total solicitada.

De forma predeterminada, ParallelCluster utiliza el escalado de listas de nodos con la mejor estrategia de lanzamiento de EC2 para lanzar algunas instancias, pero no necesariamente todas, necesarias para admitir los nodos solicitados. Intenta proporcionar la mayor capacidad posible para atender la carga de trabajo enviada.

A partir de la ParallelCluster versión 3.7.0, ParallelCluster utiliza el escalado a nivel de trabajo con una estrategia de lanzamiento de all-or-nothingEC2 para los trabajos enviados en modo exclusivo. Cuando envía un trabajo en modo exclusivo, el trabajo tiene acceso exclusivo a los nodos asignados. Para obtener más información, consulte [EXCLUSIVE](#) en la documentación de Slurm.

Para enviar un trabajo en modo exclusivo:

- Pase la marca de exclusividad al enviar un trabajo de Slurm al clúster. Por ejemplo, `sbatch ... --exclusive`.

OR

- Envíe un trabajo a una cola de clústeres que se haya configurado con el valor [JobExclusiveAllocation](#) establecido en `true`.

Al enviar un trabajo en modo exclusivo:

- ParallelCluster actualmente, agrupa las solicitudes de lanzamiento por lotes para incluir hasta 500 nodos. Si un trabajo solicita más de 500 nodos, ParallelCluster realiza una solicitud de all-or-nothinglanzamiento para cada conjunto de 500 nodos y una solicitud de lanzamiento adicional para el resto de los nodos.

- Si la asignación de nodos se realiza en un único recurso informático, ParallelCluster realiza una solicitud de all-or-nothinglanzamiento para cada conjunto de 500 nodos y una solicitud de lanzamiento adicional para el resto de los nodos. Si se produce un error en una solicitud de lanzamiento, ParallelCluster se cancela la capacidad no utilizada creada por todas las solicitudes de lanzamiento.
- Si la asignación de nodos abarca varios recursos informáticos, ParallelCluster debe realizar una solicitud de all-or-nothinglanzamiento para cada recurso informático. Estas solicitudes también se agrupan en lotes. Si se produce un error en una solicitud de lanzamiento para uno de los recursos informáticos, ParallelCluster se cancela la capacidad no utilizada creada por todas las solicitudes de lanzamiento de los recursos informáticos.

escalamiento a nivel de trabajo con limitaciones conocidas de la estrategia de all-or-nothinglanzamiento:

- Al enviar un trabajo en un recurso informático con un solo tipo de instancia, en una cola que abarca varias zonas de disponibilidad, la llamada a la API de lanzamiento de all-or-nothingEC2 solo se realiza correctamente si se puede proporcionar toda la capacidad en una sola zona de disponibilidad.
- Al enviar un trabajo en un recurso informático con varios tipos de instancias, en una cola con una única zona de disponibilidad, la llamada a la API de lanzamiento de all-or-nothingEC2 solo se realiza correctamente si un único tipo de instancia puede proporcionar toda la capacidad.
- Cuando envías un trabajo en un recurso informático con varios tipos de instancias, en una cola que abarca varias zonas de disponibilidad, no se admite la llamada a la API de lanzamiento de all-or-nothingEC2 y, en cambio, se realiza el escalado al máximo. ParallelCluster

Slurmestrategias de asignación dinámica de nodos en la versión 3.6.x y anteriores

AWS ParallelCluster utiliza solo un tipo de estrategia de asignación dinámica de nodos para escalar el clúster:

- Asignación basada en la información de nodos solicitada disponible:
 - Reanudación de todos los nodos o escalado de la lista de nodos: cuando Slurm se ejecuta, ParallelCluster amplía el clúster basándose únicamente en los nombres Slurm de la lista de nodos solicitada. ResumeProgram Asigna recursos de computación a los nodos solo por nombre de nodo. La lista de nombres de nodos puede abarcar varios trabajos.
- Asignación con una estrategia de lanzamiento de EC2:

- Escalado óptimo: ParallelCluster amplía el clúster mediante una llamada a la API de una instancia de lanzamiento de EC2 con una capacidad objetivo mínima igual a 1, para lanzar algunas instancias, pero no necesariamente todas, necesarias para admitir los nodos solicitados.

ParallelCluster utiliza el escalado de listas de nodos con la estrategia de lanzamiento de EC2 más eficaz para lanzar algunas instancias, aunque no necesariamente todas, necesarias para dar soporte a los nodos solicitados. Intenta proporcionar la mayor capacidad posible para atender la carga de trabajo enviada.

Limitaciones

- Es posible que las instancias estén inactivas al final del proceso de escalado, en el caso de que no sea posible asignar todos los nodos solicitados por las tareas.

Slurmcontabilidad con AWS ParallelCluster

A partir de la versión 3.3.0, AWS ParallelCluster admite la Slurm contabilidad con el parámetro de configuración del clúster [SlurmSettings/Base de datos](#).

Con la contabilidad de Slurm, puede integrar una base de datos contable externa para hacer lo siguiente:

- Administre los usuarios o grupos de usuarios del clúster y otras entidades. Con esta capacidad, puede utilizar las características más avanzadas de Slurm, como la aplicación de límites de recursos, la distribución equitativa y QoS.
- Recopile y guarde datos del trabajo, como el usuario que lo ejecutó, la duración del trabajo y los recursos que utiliza. Puede ver los datos guardados con la utilidad `sacct`.

Note

AWS ParallelCluster admite la Slurm contabilidad de los [servidores de bases de datos MySQL Slurm compatibles](#).


Trabajando con la Slurm contabilidad en AWS ParallelCluster

Antes de configurar la contabilidad de Slurm, debe disponer de un servidor de base de datos externo y de una base de datos que utilice el protocolo `mysql`.

Para configurar la Slurm contabilidad con AWS ParallelCluster, debe definir lo siguiente:


- La URI del servidor de base de datos externo en [Database/Uri](#). El servidor debe existir y ser accesible desde el nodo principal.
- Credenciales para acceder a la base de datos externa que se definen en [Base de datos/PasswordSecretArn](#) y [Base de datos/UserName](#). AWS ParallelCluster utiliza esta información para configurar la contabilidad a Slurm nivel y el `slurmdbd` servicio en el nodo principal. `slurmdbd` es el daemon que gestiona la comunicación entre el clúster y el servidor de la base de datos.

Para ver un tutorial, consulte [Crear un clúster con contabilidad Slurm](#).

 Note

AWS ParallelCluster realiza un arranque básico de la base de datos de Slurm contabilidad al configurar el usuario predeterminado del clúster como administrador de la Slurm base de datos. AWS ParallelCluster no añade ningún otro usuario a la base de datos de contabilidad. El cliente debe encargarse de administrar las entidades de contabilidad de la base de datos de Slurm.

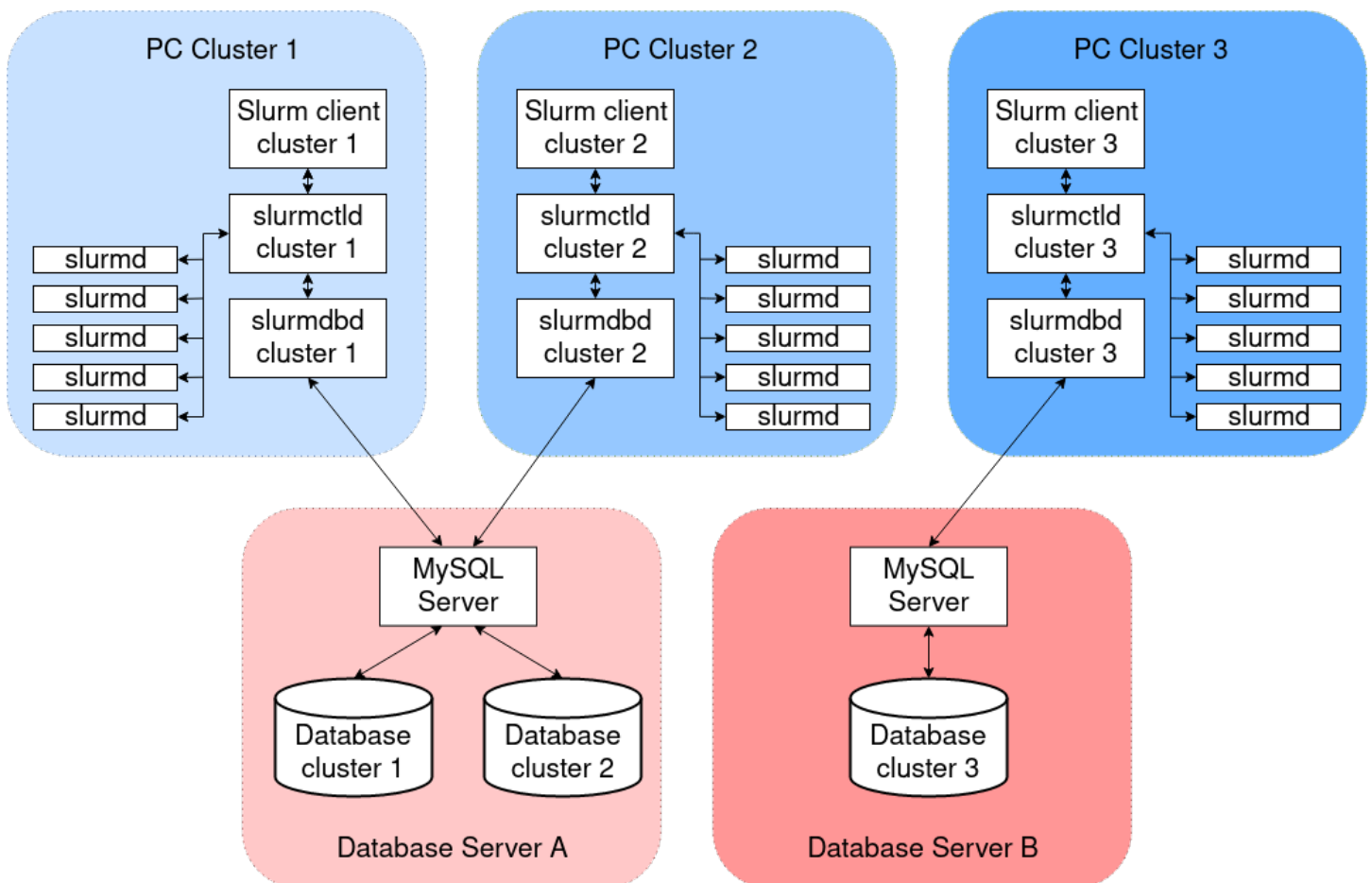
AWS ParallelCluster se configura `slurmdbd` para garantizar que un clúster tenga su propia Slurm base de datos en el servidor de bases de datos. El mismo servidor de base de datos se puede utilizar en varios clústeres, pero cada clúster tiene su propia base de datos independiente. AWS ParallelCluster usa el nombre del clúster para definir el nombre de la base de datos en el `StorageLoc` parámetro del archivo de `slurmdbd` configuración. Tenga en cuenta la siguiente situación. Una base de datos que está presente en el servidor de bases de datos incluye un nombre de clúster que no se corresponde con un nombre de clúster activo. En este caso, puede crear un clúster nuevo con ese nombre para asignarlo a esa base de datos. Slurm reutiliza la base de datos para el nuevo clúster.

 Warning

- No recomendamos configurar más de un clúster para usar la misma base de datos a la vez. Si lo hace, puede provocar problemas de rendimiento o incluso situaciones de bloqueo de la base de datos.

- Si la contabilidad de Slurm está habilitada en el nodo principal de un clúster, recomendamos usar un tipo de instancia con una CPU potente, más memoria y un mayor ancho de banda de la red. La contabilidad de Slurm puede aumentar la presión sobre el nodo principal del clúster.

En la arquitectura actual de la función de AWS ParallelCluster Slurm contabilidad, cada clúster tiene su propia instancia del `slurmdbd` daemon, como se muestra en el siguiente diagrama de configuraciones de ejemplo.



Si va a añadir funcionalidades personalizadas de varios clústeres o de federación de Slurm a su entorno de clústeres, todos los clústeres deben hacer referencia a la misma instancia de `slurmdbd`. Como alternativa, le recomendamos que habilite la AWS ParallelCluster Slurm contabilidad en un clúster y que configure manualmente los demás clústeres para que se conecten a los `slurmdbd` que están alojados en el primer clúster.

Si utilizas AWS ParallelCluster versiones anteriores a la versión 3.3.0, consulta el método alternativo para implementar la Slurm contabilidad que se describe en esta entrada del [blog sobre HPC](#).

Consideraciones sobre la contabilidad de Slurm

Base de datos y clúster en diferentes VPC

Para habilitar la contabilidad de Slurm, se necesita un servidor de base de datos que sirva de backend para las operaciones de lectura y escritura que realiza el daemon `slurmdbd`. Antes de crear o actualizar el clúster para habilitar la contabilidad de Slurm, el nodo principal debe poder acceder al servidor de base de datos.

Si necesita implementar el servidor de base de datos en una VPC distinta de la que usa el clúster, tenga en cuenta lo siguiente:

- Para habilitar la comunicación entre el `slurmdbd` del clúster y el servidor de bases de datos, debe configurar la conectividad entre las dos VPC. Para obtener más información, consulte [Interconexión de VPC](#) en la Guía del usuario de Amazon Virtual Private Cloud.
- Debe crear el grupo de seguridad que desee asociar al nodo principal de la VPC del clúster. Una vez emparejadas las dos VPC, está disponible la creación de enlaces cruzados entre los grupos de seguridad de la base de datos y los del clúster. Para obtener más información, consulte [Reglas del grupo de seguridad](#) en la Guía del usuario de Amazon Virtual Private Cloud.

Configuración del cifrado TLS entre `slurmdbd` y el servidor de la base de datos

Con la configuración de Slurm contabilidad predeterminada que se AWS ParallelCluster proporciona, `slurmdbd` establece una conexión cifrada con TLS con el servidor de la base de datos, si el servidor admite el cifrado con TLS. AWS servicios de bases de datos como Amazon RDS y Amazon Aurora admiten el cifrado TLS de forma predeterminada.

Puede requerir conexiones seguras en el servidor configurando el parámetro `require_secure_transport` en el servidor de la base de datos. Esto se configura en la plantilla proporcionada CloudFormation .

Siguiendo las prácticas recomendadas de seguridad, le recomendamos que habilite también la verificación de identidad del servidor en el cliente de `slurmdbd`. Para ello, configure el [StorageParameters](#) `enslurmdbd.conf`. Cargue el certificado de entidad de certificación del servidor en el nodo principal del clúster. A continuación, defina la opción [SSL_CA](#) de `StorageParameters`

en `slurmdbd.conf` como la ruta del certificado de entidad de certificación del servidor en el nodo principal. De este modo, se habilita la verificación de la identidad del servidor en `slurmdbd`. Tras realizar estos cambios, reinicie el servicio `slurmdbd` para volver a establecer la conectividad con el servidor de la base de datos con la verificación de identidad habilitada.

Actualización de credenciales de la base de datos

Para actualizar los valores de [Database/UserName](#) o [PasswordSecretArn](#), primero debe detener la flota de cómputo. Supongamos que el valor secreto que está almacenado en el AWS Secrets Manager secreto cambia y su ARN no cambia. En esta situación, el clúster no actualiza automáticamente la contraseña de la base de datos con el nuevo valor. Para actualizar el clúster con el nuevo valor secreto, ejecute el siguiente comando desde el nodo principal.

```
$ sudo /opt/parallelcluster/scripts/slurm/update_slurm_database_password.sh
```

Warning

Para evitar perder datos de contabilidad, le recomendamos que cambie únicamente la contraseña de la base de datos cuando la flota de computación esté detenida.

Supervisión de la base de datos

Se recomienda activar las funciones de supervisión de los servicios de AWS base de datos. Para obtener más información, consulte la documentación sobre el [monitoreo de Amazon RDS](#) o el [monitoreo de Amazon Aurora](#).

Personalización de la configuración de Slurm

A partir de AWS ParallelCluster versión 3.6.0, puede personalizar la `slurm.conf` configuración de Slurm en una configuración de clúster AWS ParallelCluster.

En la configuración del clúster, puede personalizar los parámetros Slurm de configuración mediante los siguientes ajustes de configuración del clúster:

- Personalice los parámetros de Slurm de todo el clúster mediante el parámetro [SlurmSettings / CustomSlurmSettings](#) o el parámetro [CustomSlurmSettingsIncludeFile](#). AWS ParallelCluster falla si se especifican ambos.

- Personalice los parámetros de Slurm de una cola mediante [SlurmQueues](#) / [CustomSlurmSettings](#) (asignado a las particiones Slurm).
- Personalice los parámetros de Slurm de un recurso informático mediante [SlurmQueues](#) / [ComputeResources](#) / [CustomSlurmSettings](#) (mapeados a los nodos Slurm).

Límites de personalización de la configuración de Slurm y consideraciones a tener en cuenta al usar AWS ParallelCluster

- En cuanto a la configuración `CustomSlurmSettings` y `CustomSlurmSettingsIncludeFile`, solo puede especificar y actualizar los parámetros de `slurm.conf` que estén incluidos en la [Slurmversión](#) compatible con la versión AWS ParallelCluster que esté utilizando para configurar un clúster.
- Si especifica configuraciones de Slurm personalizadas en alguno de los parámetros de `CustomSlurmSettings`, AWS ParallelCluster realiza comprobaciones de validación e impide establecer o actualizar los parámetros de configuración de Slurm que entren en conflicto con la lógica AWS ParallelCluster. Los parámetros de configuración de Slurm con los que se sabe que entran en conflicto con AWS ParallelCluster se identifican en las listas de denegación. Las listas de denegación pueden cambiar en futuras versiones de AWS ParallelCluster si se añaden otras funciones de Slurm. Para obtener más información, consulte [Parámetros de configuración de Slurm enumerados en denegación para CustomSlurmSettings](#).
- AWS ParallelCluster solo comprueba si un parámetro está en una lista de denegaciones. AWS ParallelCluster no valida la sintaxis ni la semántica de los parámetros de configuración personalizados de Slurm. Usted es responsable de validar los parámetros de configuración de Slurm personalizados. Los parámetros de configuración personalizados de Slurm no válidos pueden provocar errores en los daemons de Slurm y provocar errores en la creación y actualización del clúster.
- Si especifica configuraciones personalizadas de Slurm en `CustomSlurmSettingsIncludeFile`, AWS ParallelCluster no realiza ninguna validación.
- Puede actualizar `CustomSlurmSettings` y `CustomSlurmSettingsIncludeFile` sin detener ni iniciar la flota de cómputo. En este caso, AWS ParallelCluster reinicia el daemon de `slurmctld` y ejecuta el comando `scontrol reconfigure`.

Es posible que algunos parámetros de configuración de Slurm requieran operaciones diferentes antes de que se registre un cambio en todo el clúster. Por ejemplo, es posible que requieran el reinicio de todos los daemons del clúster. Usted es responsable de comprobar

si las operaciones del AWS ParallelCluster son suficientes para propagar los parámetros de configuración personalizados de Slurm durante las actualizaciones. Si considera que las operaciones de AWS ParallelCluster no son suficientes, es su responsabilidad realizar las acciones adicionales necesarias para propagar la configuración actualizada, tal como se recomienda en la [Slurmdocumentación](#).

Parámetros de configuración de Slurm enumerados en denegación para **CustomSlurmSettings**

En las tablas siguientes se enumeran los parámetros con las versiones de AWS ParallelCluster que deniegan su uso, empezando por la versión 3.6.0. CustomSlurmSettings no es compatible con versiones de AWS ParallelCluster anteriores a la 3.6.0.

Parámetros de la lista de denegación a nivel de clúster:

Parámetro Slurm	Listado de denegación en las versiones de AWS ParallelCluster
CommunicationParameters	3.6.0
Epilog	3.6.0
GresTypes	3.6.0
LaunchParameters	3.6.0
Prolog	3.6.0
ReconfigFlags	3.6.0
ResumeFailProgram	3.6.0
ResumeProgram	3.6.0
ResumeTimeout	3.6.0
SlurmctldHost	3.6.0
SlurmctldLogFile	3.6.0
SlurmctldParameters	3.6.0

Parámetro Slurm	Listado de denegación en las versiones de AWS ParallelCluster
SlurmdLogfile	3.6.0
SlurmUser	3.6.0
SuspendExcNodes	3.6.0
SuspendProgram	3.6.0
SuspendTime	3.6.0
TaskPlugin	3.6.0
TreeWidth	3.6.0

Parámetros de lista denegados a nivel de clúster cuando la [integración de contabilidad nativa de Slurm](#) está configurada en la configuración del clúster:

Parámetro Slurm	Listado de denegados en las versiones AWS ParallelCluster
AccountingStorageType	3.6.0
AccountingStorageHost	3.6.0
AccountingStoragePort	3.6.0
AccountingStorageUser	3.6.0
JobAcctGatherType	3.6.0

Parámetros de lista denegados a nivel de cola (partición) para las colas gestionadas por AWS ParallelCluster:

Parámetro Slurm	Listado de denegados en las versiones AWS ParallelCluster
Nodos	3.6.0
PartitionName	3.6.0
ResumeTimeout	3.6.0
State	3.6.0
SuspendTime	3.6.0

Parámetros de lista denegados a nivel de recurso de cómputo (nodo) para un recurso de cómputo gestionado por AWS ParallelCluster:

Parámetro Slurm	Incluido en la versión denegada de la versión AWS ParallelCluster y en las versiones posteriores
CPUs	3.6.0
Características	3.6.0
Gres	3.6.0
NodeAddr	3.6.0
NodeHostname	3.6.0
NodeName	3.6.0
Weight	3.7.0

Slurm **prolog** y **epilog**

A partir de AWS ParallelCluster versión 3.6.0, la configuración Slurm que se implementa con AWS ParallelCluster incluye los siguientes parámetros de configuración Prolog y Epilog:

```
# PROLOG AND EPILOG
Prolog=/opt/slurm/etc/scripts/prolog.d/*
Epilog=/opt/slurm/etc/scripts/epilog.d/*
SchedulerParameters=nohold_on_prolog_fail
BatchStartTimeout=180
```

Para obtener más información, consulte la [Guía de Prólogo y Epílogo](#) en la Slurm documentación.

AWS ParallelCluster incluye los siguientes guiones de prólogo y epílogo:

- `90_plcluster_health_check_manager` (en la carpeta Prolog)
- `90_pcluster_noop` (en la carpeta Epilog)

Note

Tanto la carpeta Prolog como la Epilog deben contener un archivo como mínimo.

Puede utilizar sus propios scripts prolog o epilog scripts añadiéndolos a las carpetas Prolog y Epilog correspondientes.

Warning

Slurm ejecuta todos los scripts de las carpetas, en orden alfabético inverso.

La duración del tiempo de ejecución de los scripts prolog y epilog afecta al tiempo necesario para ejecutar un trabajo. Actualice el ajuste BatchStartTimeout de configuración cuando ejecute varios scripts o scripts prolog de ejecución prolongada. El valor predeterminado es de 3 minutos.

Si utiliza scripts prolog y epilog personalizados, localice los scripts en las carpetas Prolog y Epilog correspondientes. Le recomendamos que mantenga el script `90_plcluster_health_check_manager` que se ejecuta antes de cada script personalizado. Para obtener más información, consulte [Personalización de la configuración de Slurm](#).

Tamaño y actualización de la capacidad del clúster

La capacidad del clúster se define por la cantidad de nodos de procesamiento que el clúster puede escalar. Los nodos de cómputo están respaldados por instancias de EC2 definidas dentro de los recursos de cómputo de la AWS ParallelCluster configuración ([Scheduling/SlurmQueues/ComputeResources](#)) y están organizados en colas ([Scheduling/SlurmQueues](#)) que se asignan a Slurm las particiones 1:1.

Dentro de un recurso de cómputo, es posible configurar el número mínimo de nodos de cómputo (instancias) que deben mantenerse siempre en ejecución en el clúster ([MinCount](#)) y el número máximo de instancias al que el recurso de cómputo puede escalar ([MaxCount3](#)).

En el momento de la creación del clúster, o tras una actualización del clúster, AWS ParallelCluster lanza tantas instancias de EC2 como estén configuradas `MinCount` para cada recurso informático ([Scheduling/SlurmQueues/ ComputeResources](#)) definido en el clúster. Las instancias lanzadas para cubrir la cantidad mínima de nodos de un recurso informático del clúster se denominan nodos estáticos. Una vez iniciados, los nodos estáticos deben permanecer en el clúster y el sistema no los termina, a menos que se produzca un evento o una condición en particular. Estos eventos incluyen, por ejemplo, el fallo de las comprobaciones de estado de EC2 y el cambio del estado del nodo Slurm a DRAIN Slurm o DOWN.

Las instancias EC2, que se lanzan bajo demanda **1** para **'MaxCount - MinCount'** hacer frente al aumento de carga del clúster, se denominan nodos dinámicos. **MaxCount MinCount**) Su naturaleza es efímera: se lanzan para atender tareas pendientes y se cancelan una vez que permanecen inactivas durante un período de tiempo definido [Scheduling/SlurmSettings/ScaledownIdleTime](#) en la configuración del clúster (predeterminado: 10 minutos).

Los nodos estáticos y los nodos dinámicos cumplen con el siguiente esquema de nomenclatura:

- Nodos estáticos `<Queue/Name>-st-<ComputeResource/Name>-<num>` donde `<num> = 1..ComputeResource/MinCount`
- `<Queue/Name>-dy-<ComputeResource/Name>-<num>` ¿Dónde están los nodos dinámicos? `<num> = 1..(ComputeResource/MaxCount - ComputeResource/MinCount)`

Por ejemplo, dada la siguiente AWS ParallelCluster configuración:

```
Scheduling:
```

```
Scheduler: slurm
SlurmQueues:
  - Name: queue1
    ComputeResources:
      - Name: c5xlarge
        Instances:
          - InstanceType: c5.xlarge
            MinCount: 100
            MaxCount: 150
```

Los siguientes nodos se definirán en Slurm

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up      infinite   50    idle~ queue1-dy-c5xlarge-[1-50]
queue1*   up      infinite  100    idle queue1-st-c5xlarge-[1-100]
```

Cuando un recurso de cómputo lo tenga `MinCount == MaxCount`, todos los nodos de cómputo correspondientes serán estáticos y todas las instancias se lanzarán en el momento de la creación o actualización del clúster y se mantendrán en funcionamiento. Por ejemplo:

```
Scheduling:
Scheduler: slurm
SlurmQueues:
  - Name: queue1
    ComputeResources:
      - Name: c5xlarge
        Instances:
          - InstanceType: c5.xlarge
            MinCount: 100
            MaxCount: 100
```

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up      infinite   100    idle queue1-st-c5xlarge-[1-100]
```

Actualización de la capacidad del clúster

La actualización de la capacidad del clúster incluye añadir o eliminar colas, recursos informáticos o cambiar la capacidad MinCount/MaxCount de un recurso informático. A partir de la AWS ParallelCluster versión 3.9.0, para reducir el tamaño de una cola es necesario detener la flota informática o [QueueUpdateStrategy](#) configurarla en TERMINATE antes de que se lleve a cabo la actualización del clúster. No es necesario detener la flota de cómputo ni configurarla en TERMINATE [QueueUpdateStrategy](#) cuando:

- Añadir nuevas colas a Scheduling/ [SlurmQueues](#)
- Añadir nuevos recursos de cómputo a una cola Scheduling/ [SlurmQueues/ComputeResources](#)
- Aumentar la cantidad [MaxCount](#) de un recurso de cómputo
- Incremento MinCount de un recurso informático y aumento MaxCount del mismo recurso informático en al menos la misma cantidad

Consideraciones y limitaciones

El objetivo de esta sección es describir los factores, restricciones o limitaciones importantes que se deben tener en cuenta al redimensionar la capacidad del clúster.

- Al eliminar una cola de Scheduling/<https://docs.aws.amazon.com/parallelcluster/latest/ug/Scheduling-v3.html#Scheduling-v3-SlurmQueues> SlurmQueues todos los nodos de procesamiento con un nombre <Queue/Name> - *, tanto estático como dinámico, se eliminarán de la Slurm configuración y se eliminarán las instancias de EC2 correspondientes.
- Al eliminar un recurso informático Scheduling/SlurmQueues/<https://docs.aws.amazon.com/parallelcluster/latest/ug/Scheduling-v3.html#Scheduling-v3-SlurmQueues-ComputeResources> ComputeResources de una cola, todos los nodos informáticos con nombre <Queue/Name> - * - <ComputeResource/Name> - *, tanto estáticos como dinámicos, se eliminarán de la Slurm configuración y se eliminarán las instancias de EC2 correspondientes.

Al cambiar el MinCount parámetro de un recurso informático, podemos distinguir dos escenarios diferentes: si MaxCount se mantiene igual a MinCount (solo capacidad estática) y si MaxCount es mayor que MinCount (capacidad estática y dinámica mixta).

La capacidad cambia solo con los nodos estáticos

- Si `MinCount == MaxCount`, al aumentar `MinCount` (y `MaxCount`), el clúster se configurará ampliando el número de nodos estáticos hasta el nuevo valor de `MinCount` `<Queue/Name>-st-<ComputeResource/Name>-<new_MinCount>` y el sistema seguirá intentando lanzar instancias EC2 para cumplir con la nueva capacidad estática requerida.
- Si `MinCount == MaxCount`, al disminuir `MinCount` (y `MaxCount`) la cantidad `N`, el clúster se configurará eliminando los últimos `N` nodos estáticos `<Queue/Name>-st-<ComputeResource/Name>-<old_MinCount - N>...<old_MinCount>` y el sistema cerrará las instancias de EC2 correspondientes.
- Estado inicial `MinCount = MaxCount = 100`

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up       infinite   100   idle queue1-st-c5xlarge-[1-100]
```

- Actualización `-30` en `MinCount` y `MaxCount`: `MinCount = MaxCount = 70`

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up       infinite    70   idle queue1-st-c5xlarge-[1-70]
```

La capacidad cambia con nodos mixtos

Si `MinCount < MaxCount`, al aumentar `MinCount` en una cantidad `N` (suponiendo que `MaxCount` se mantenga sin cambios), el clúster se configurará ampliando el número de nodos estáticos hasta el nuevo valor de `MinCount` (`old_MinCount + N`): `<Queue/Name>-st-<ComputeResource/Name>-<old_MinCount + N>` y el sistema seguirá intentando lanzar instancias EC2 para cumplir con la nueva capacidad estática requerida. Además, para respetar la `MaxCount` capacidad del recurso informático, la configuración del clúster se actualiza eliminando los últimos `N` nodos dinámicos, `<Queue/Name>-dy-<ComputeResource/Name>-[<MaxCount - old_MinCount - N>...<MaxCount - old_MinCount>]` y el sistema finalizará las instancias de EC2 correspondientes.

- Estado inicial: MinCount = 100; MaxCount = 150

```

$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up    infinite   50    idle~ queue1-dy-c5xlarge-[1-50]
queue1*    up    infinite  100    idle queue1-st-c5xlarge-[1-100]

```

- Actualiza +30 a MinCount : MinCount = 130 (MaxCount = 150)

```

$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up    infinite   20    idle~ queue1-dy-c5xlarge-[1-20]
queue1*    up    infinite  130    idle queue1-st-c5xlarge-[1-130]

```

Si $\text{MinCount} < \text{MaxCount}$, al aumentar N MinCount y en MaxCount la misma cantidad, el clúster se configurará ampliando el número de nodos estáticos hasta el nuevo valor de MinCount ($\text{old_MinCount} + N$): `<Queue/Name>-st-<ComputeResource/Name>-<old_MinCount + N>` y el sistema seguirá intentando lanzar instancias EC2 para cumplir con la nueva capacidad estática requerida. Además, no se realizará ningún cambio en el número de nodos dinámicos para adaptarlo a los nuevos

Valor MaxCount.

- Estado inicial: MinCount = 100; MaxCount = 150

```

$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up    infinite   50    idle~ queue1-dy-c5xlarge-[1-50]
queue1*    up    infinite  100    idle queue1-st-c5xlarge-[1-100]

```

- Actualiza +30 a MinCount : MinCount = 130 (MaxCount = 180)

```

$ sinfo

```

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up    infinite    20    idle~ queue1-dy-c5xlarge-[1-50]
queue1*    up    infinite   130    idle queue1-st-c5xlarge-[1-130]
```

Si $\text{MinCount} < \text{MaxCount}$, al reducir la MinCount cantidad N (suponiendo que MaxCount se mantenga sin cambios), el clúster se configurará eliminando los últimos N nodos estáticos $\langle \text{Queue/Name} \rangle\text{-st-}\langle \text{ComputeResource/Name} \rangle\text{-}[\langle \text{old_MinCount} \rangle - N \dots \langle \text{old_MinCount} \rangle]$ y el sistema cerrará las instancias de EC2 correspondientes. Además, para respetar la MaxCount capacidad del recurso informático, la configuración del clúster se actualiza ampliando el número de nodos dinámicos para cubrir el vacío. $\text{MaxCount} - \text{new_MinCount}$: $\langle \text{Queue/Name} \rangle\text{-dy-}\langle \text{ComputeResource/Name} \rangle\text{-}[1 \dots \langle \text{MaxCount} \rangle - \text{new_MinCount}]$ En este caso, dado que se trata de nodos dinámicos, no se lanzará ninguna nueva instancia de EC2 a menos que el planificador tenga tareas pendientes en los nuevos nodos.

- Estado inicial: $\text{MinCount} = 100$; $\text{MaxCount} = 150$

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up    infinite    50    idle~ queue1-dy-c5xlarge-[1-50]
queue1*    up    infinite   100    idle queue1-st-c5xlarge-[1-100]
```

- Actualización -30 el MinCount : $\text{MinCount} = 70$ ($\text{MaxCount} = 120$)

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up    infinite    80    idle~ queue1-dy-c5xlarge-[1-80]
queue1*    up    infinite    70    idle queue1-st-c5xlarge-[1-70]
```

Si $\text{MinCount} < \text{MaxCount}$, al disminuir MinCount y con MaxCount la misma cantidad N , el clúster se configurará eliminando los últimos N nodos estáticos $\langle \text{Queue/Name} \rangle\text{-st-}\langle \text{ComputeResource/}$

Name>-<old_MinCount - N>...<oldMinCount>] y el sistema finalizará las instancias de EC2 correspondientes.

Además, no se realizará ningún cambio en el número de nodos dinámicos para respetar el nuevo MaxCount valor.

- Estado inicial: MinCount = 100; MaxCount = 150

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up    infinite   50    idle~ queue1-dy-c5xlarge-[1-50]
queue1*    up    infinite  100    idle queue1-st-c5xlarge-[1-100]
```

- Actualización -30 el MinCount : MinCount = 70 (MaxCount = 120)

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up    infinite   80    idle~ queue1-dy-c5xlarge-[1-50]
queue1*    up    infinite   70    idle queue1-st-c5xlarge-[1-70]
```

Si $\text{MinCount} < \text{MaxCount}$, al disminuir la MaxCount cantidad N (suponiendo que MinCount se mantenga sin cambios), el clúster se configurará eliminando los últimos N nodos dinámicos <Queue/Name>-dy-<ComputeResource/Name>-<old_MaxCount - N...<oldMaxCount>] y el sistema cerrará las instancias de EC2 correspondientes en caso de que estuvieran en ejecución. No se espera ningún impacto en los nodos estáticos.

- Estado inicial: MinCount = 100; MaxCount = 150

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*    up    infinite   50    idle~ queue1-dy-c5xlarge-[1-50]
queue1*    up    infinite  100    idle queue1-st-c5xlarge-[1-100]
```

- Actualización -30 el MaxCount : MinCount = 100 (MaxCount = 120)

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
queue1*   up    infinite    20   idle~ queue1-dy-c5xlarge-[1-20]
queue1*   up    infinite   100   idle  queue1-st-c5xlarge-[1-100]
```

Impactos en los puestos de trabajo

En todos los casos en los que se eliminen los nodos y se cierren las instancias de EC2, se volverá a poner en cola un trabajo por lotes que se ejecute en los nodos eliminados, a menos que no haya otros nodos que cumplan con los requisitos del trabajo. En este último caso, el trabajo fallará con el estado `NODE_FAIL` y desaparecerá de la cola; si es el caso, será necesario volver a enviarlo manualmente.

Si planea realizar una actualización de tamaño del clúster, puede impedir que los trabajos se ejecuten en los nodos que se van a eliminar durante la actualización planificada. Esto es posible configurando los nodos que se eliminarán durante el mantenimiento. Tenga en cuenta que poner un nodo en fase de mantenimiento no afectaría a los trabajos que eventualmente ya se estén ejecutando en el nodo.

Supongamos que con la actualización planificada para cambiar el tamaño del clúster [se va a eliminar el nodo `queueu-st-computeresource-[9-10]`]. Puede crear una Slurm reserva con el siguiente comando

```
sudo -i scontrol create reservation ReservationName=maint_for_update user=root
starttime=now duration=infinite flags=maint,ignore_jobs nodes=queueu-st-
computeresource-[9-10]
```

Esto creará una Slurm reserva con un nombre `maint_for_update` en los nodos `queueu-st-computeresource-[9-10]`. Desde el momento en que se crea la reserva, no se pueden ejecutar más trabajos en los nodos `queueu-st-computeresource-[9-10]`. Tenga en cuenta que la reserva no impedirá que los trabajos se asignen eventualmente a los nodos `queueu-st-computeresource-[9-10]`.

Tras la actualización del tamaño del clúster, si la Slurm reserva se estableció solo en los nodos que se eliminaron durante la actualización del cambio de tamaño, la reserva de mantenimiento se

eliminará automáticamente. Si, por el contrario, ha creado una Slurm reserva en los nodos que siguen presentes tras la actualización del cambio de tamaño del clúster, es posible que deseemos eliminar la reserva de mantenimiento de los nodos una vez realizada la actualización de tamaño mediante el siguiente comando

```
sudo -i scontrol delete ReservationName=maint_for_update
```

[Para obtener más información sobre la Slurm reserva, consulta el documento oficial de SchedMD aquí.](#)

Proceso de actualización del clúster sobre los cambios de capacidad

Tras un cambio en la configuración del programador, se ejecutan los siguientes pasos durante el proceso de actualización del clúster:

- Detenga AWS ParallelCluster `clustermgtd` (`supervisorctl stop clustermgtd`)
- Genere la configuración de Slurm particiones actualizada a partir de AWS ParallelCluster la configuración
- Reiniciar `slurmctld` (realizado mediante la receta del servicio Chef)
- Compruebe el `slurmctld` estado (`systemctl is-active --quiet slurmctld.service`)
- Recargar la configuración Slurm (`scontrol reconfigure`)
- Iniciar `clustermgtd` (`supervisorctl start clustermgtd`)

AWS Batch (**awsbatch**)

Para obtener más información sobre AWS Batch, consulte [AWS Batch](#). Para ver la documentación, consulte la [Guía del usuario de AWS Batch](#).

Comandos de la CLI de AWS ParallelCluster para AWS Batch

Cuando usa el programador `awsbatch`, los comandos de la CLI de AWS ParallelCluster para AWS Batch se instalan automáticamente en el nodo principal de AWS ParallelCluster. La CLI usa operaciones de la API de AWS Batch y permite las siguientes operaciones:

- Enviar y administrar trabajos
- Monitorear trabajos, colas y hosts
- Crear un reflejo de los comandos del programador tradicionales

⚠ Important

AWS ParallelCluster no admite trabajos de GPU para AWS Batch. Para obtener más información, consulte [Trabajos de la GPU](#).

Esta CLI se distribuye como un paquete independiente. Para obtener más información, consulte [Compatibilidad con el programador](#).

Temas

- [awsbsub](#)
- [awsbstat](#)
- [awsbout](#)
- [awsbkill](#)
- [awsbqueues](#)
- [awsbhosts](#)

awsbsub

Envía trabajos a la cola de trabajos del clúster.

```
awsbsub [-h] [-jn JOB_NAME] [-c CLUSTER] [-cf] [-w WORKING_DIR]  
        [-pw PARENT_WORKING_DIR] [-if INPUT_FILE] [-p VCPUS] [-m MEMORY]  
        [-e ENV] [-eb ENV_DENYLIST] [-r RETRY_ATTEMPTS] [-t TIMEOUT]  
        [-n NODES] [-a ARRAY_SIZE] [-d DEPENDS_ON]  
        [command] [arguments [arguments ...]]
```

⚠ Important

AWS ParallelCluster no admite trabajos de GPU para AWS Batch. Para obtener más información, consulte [Trabajos de la GPU](#).

Argumentos de posición

command

Envía el trabajo (el comando especificado debe estar disponible en las instancias de computación) o el nombre de archivo que se va a transferir. Véase también `--command-file`.

arguments

(Opcional) Especifica argumentos para el comando o el archivo de comandos.

Argumentos con nombre

-jn *JOB_NAME*, --job-name *JOB_NAME*

Asigna un nombre al trabajo. El primer carácter debe ser una letra o un número. El nombre del trabajo puede contener letras (mayúsculas y minúsculas), números, guiones (-) y guiones bajos (_), y puede tener una longitud máxima de 128 caracteres.

-c *CLUSTER*, --cluster *CLUSTER*

Especifica el clúster que se va a utilizar.

-cf, --command-file

Indica que el comando es un archivo que se va a transferir a las instancias de informática.

Valor predeterminado: False

-w *WORKING_DIR*, --working-dir *WORKING_DIR*

Especifica la carpeta que se va a usar como directorio de trabajo del trabajo. Si no se especifica un directorio de trabajo, el trabajo se ejecuta en la subcarpeta `job-<AWS_BATCH_JOB_ID>` del directorio de inicio del usuario. Puede utilizar este parámetro o el parámetro `--parent-working-dir`.

-pw *PARENT_WORKING_DIR*, --parent-working-dir *PARENT_WORKING_DIR*

Especifica la carpeta principal del directorio de trabajo del trabajo. Si no se especifica un directorio de trabajo principal, el valor predeterminado es el directorio de inicio del usuario. Una subcarpeta llamada `job-<AWS_BATCH_JOB_ID>` se crea en el directorio de trabajo principal. Puede utilizar este parámetro o el parámetro `--working-dir`.

-if *INPUT_FILE*, --input-file *INPUT_FILE*

Especifica el archivo que se va a transferir a las instancias de computación, en el directorio de trabajo de la tarea. Puede especificar varios parámetros de archivo de entrada.

-p *VCPUS*, --vcpus *VCPUS*

Especifica la cantidad de CPU virtuales que quiere reservar para el contenedor. Cuando se utiliza junto con `-nodes`, identifica la cantidad de vCPU por cada nodo.

Predeterminado: 1

-m *MEMORY*, --memory *MEMORY*

Especifica el límite máximo de memoria (en MiB) que se va a proporcionar para el trabajo. Si su trabajo intenta superar el límite de memoria especificado aquí, se finaliza el trabajo.

Predeterminado: 128

-e *ENV*, --env *ENV*

Especifica una lista de nombres de variable de entorno separados por comas que se van a exportar al entorno de trabajo. Para exportar todas las variables de entorno, especifique "todas". Tenga en cuenta que una lista de "todas" las variables de entorno no incluyen aquellas que figuren en el parámetro `-env-blacklist` ni tampoco variables que empiecen por el prefijo `PCLUSTER_*` o `AWS_*`.

-eb *ENV_DENYLIST*, --env-blacklist *ENV_DENYLIST*

Especifica una lista de nombres de variable de entorno separados por comas que no se van a exportar al entorno de trabajo. De forma predeterminada, no se exportan `HOME`, `PWD`, `USER`, `PATH`, `LD_LIBRARY_PATH`, `TERM` ni `TERMCAP`.

-r *RETRY_ATTEMPTS*, --retry-attempts *RETRY_ATTEMPTS*

Especifica la cantidad de veces que toma pasar un trabajo al estado `RUNNABLE`. Puede especificar entre 1 y 10 intentos. Si el valor de los intentos es superior a 1, el trabajo se reintenta si produce un error, hasta que haya pasado al estado `RUNNABLE`, que especificaba la cantidad de veces.

Predeterminado: 1

-t *TIMEOUT*, --timeout *TIMEOUT*

Especifica la duración en segundos (medida a partir de la marca de tiempo `startedAt` del intento de trabajo) después de la cual AWS Batch termina el trabajo si no ha finalizado. El valor del tiempo de inactividad debe ser al menos 60 segundos.

-n *NODES*, --nodes *NODES*

Especifica el número de nodos que quiere reservar para el trabajo. Especifique un valor para que este parámetro habilite el envío paralelo de varios nodos.

i Note

Cuando el parámetro [Scheduler/AwsBatchQueues/CapacityType](#) está establecido en SPOT, no se admiten los trabajos paralelos de varios nodos. Además, debe haber un rol vinculado a servicios `AWSServiceRoleForEC2Spot` en su cuenta. Puede crear este rol con el siguiente comando de AWS CLI:

```
$ aws iam create-service-linked-role --aws-service-name spot.amazonaws.com
```

Para obtener más información, consulte [Rol vinculado al servicio para solicitudes de instancias de spot](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

-a *ARRAY_SIZE*, --array-size *ARRAY_SIZE*

Indica el tamaño de la matriz. Puede especificar un valor comprendido entre 2 y 10 000. Si especifica las propiedades de matriz para un trabajo, pasa a ser un trabajo de matriz.

-d *DEPENDS_ON*, --depends-on *DEPENDS_ON*

Especifica una lista de dependencias separadas por puntos y coma para un trabajo. Un trabajo puede depender de un máximo de 20 trabajos. Puede especificar una dependencia de tipo SEQUENTIAL sin especificar un ID de trabajo para los trabajos de matriz. Una dependencia SEQUENTIAL permite que cada trabajo de matriz secundario se complete de forma secuencial, a partir del índice 0. También puede especificar una dependencia de tipo N_TO_N con un ID de trabajo para los trabajos de matriz. Una dependencia N_TO_N conlleva que cada índice secundario de este trabajo deba esperar a que se complete el índice secundario correspondiente de cada dependencia antes de comenzar. La sintaxis de este parámetro es “`jobId=<string>,type=<string>;...`”.

awsbstat

Muestra los trabajos que se envían en la cola de trabajos del clúster.

```
awsbstat [-h] [-c CLUSTER] [-s STATUS] [-e] [-d] [job_ids [job_ids ...]]
```

Argumentos de posición

job_ids

Especifica la lista de ID de trabajo separados por espacios que se van a mostrar en la salida. Si el trabajo es una matriz de trabajo, se muestran todos los trabajos secundarios. Si se solicita un solo trabajo, se muestra en una versión detallada.

Argumentos con nombre

-c *CLUSTER*, --cluster *CLUSTER*

Indica el clúster que se va a utilizar.

-s *STATUS*, --status *STATUS*

Especifica una lista de estados de los trabajos separados por comas que se van a incluir. El estado del trabajo predeterminado es "activo". Los valores aceptados son: SUBMITTED, PENDING, RUNNABLE, STARTING, RUNNING, SUCCEEDED, FAILED y ALL.

Valor predeterminado: "SUBMITTED,PENDING,RUNNABLE,STARTING y RUNNING"

-e, --expand-children

Amplía los trabajos con elementos secundarios (tanto los de matriz como los paralelos de varios nodos).

Valor predeterminado: False

-d, --details

Muestra detalles de los trabajos.

Valor predeterminado: False

awsbout

Muestra la salida de un trabajo especificado.

```
awsbout [-h] [-c CLUSTER] [-hd HEAD] [-t TAIL] [-s] [-sp STREAM_PERIOD] job_id
```

Argumentos de posición

job_id

Especifica el ID de trabajo.

Argumentos con nombre

-c *CLUSTER*, --cluster *CLUSTER*

Indica el clúster que se va a utilizar.

-hd *HEAD*, --head *HEAD*

Obtiene las primeras líneas *HEAD* de la salida del trabajo.

-t *TAIL*, --tail *TAIL*

Obtiene las últimas líneas <tail> de la salida del trabajo.

-s, --stream

Obtiene la salida del trabajo y, a continuación, espera a que se produzca una salida adicional. Este argumento puede usarse junto a `-tail` para comenzar desde las últimas líneas <tail> de la salida del trabajo.

Valor predeterminado: False

-sp *STREAM_PERIOD*, --stream-period *STREAM_PERIOD*

Establece el periodo de streaming.

Predeterminado: 5

awsbkill

Cancela o termina trabajos enviados en el clúster.

```
awsbkill [-h] [-c CLUSTER] [-r REASON] job_ids [job_ids ... ]
```

Argumentos de posición

job_ids

Especifica la lista de ID de trabajo separados por espacios que se deben cancelar o terminar.

Argumentos con nombre

-c *CLUSTER*, --cluster *CLUSTER*

Indica el nombre del clúster que se va a utilizar.

-r *REASON*, --reason *REASON*

Indica el mensaje que se asociará a un trabajo, en el cual se explica el motivo de su cancelación.

Valor predeterminado: "Terminated by the user"

awsbqueues

Muestra la cola de trabajos que se asocia al clúster.

```
awsbqueues [-h] [-c CLUSTER] [-d] [job_queues [job_queues ... ]]
```

Argumentos de posición

job_queues

Especifica la lista de nombres de colas separados por espacios que se van a mostrar. Si se solicita una sola cola, se muestra en una versión detallada.

Argumentos con nombre

-c *CLUSTER*, --cluster *CLUSTER*

Especifica el nombre del clúster que se va a utilizar.

-d, --details

Indica si se deben mostrar los detalles de las colas.

Valor predeterminado: False

awsbhosts

Muestra los hosts que pertenecen al entorno informático del clúster.

```
awsbhosts [-h] [-c CLUSTER] [-d] [instance_ids [instance_ids ... ]]
```

Argumentos de posición

instance_ids

Especifica una lista de ID de instancias separados por espacios. Si se solicita una sola instancia, se muestra en una versión detallada.

Argumentos con nombre

-c *CLUSTER*, **--cluster** *CLUSTER*

Especifica el nombre del clúster que se va a utilizar.

-d, **--details**

Indica si se deben mostrar los detalles de los hosts.

Valor predeterminado: False

Almacenamiento compartido

AWS ParallelCluster [admite el uso de Amazon EBS, FSx para ONTAP y FSx para los volúmenes de almacenamiento compartido OpenZFS, Amazon EFS y FSx para los sistemas de archivos de almacenamiento compartido Lustre o caché de archivos](#). Le recomendamos que siga las pautas básicas de [confiabilidad de un marco AWS bien diseñado y que haga copias de seguridad de sus volúmenes](#) y sistemas de archivos.

Seleccione un sistema de almacenamiento que cumpla con los requisitos de E/S de sus aplicaciones de HPC. Puede optimizar cada sistema de archivos en función de su caso de uso específico. Para obtener más información, consulte [Resumen de opciones de almacenaje](#).

Los volúmenes de Amazon EBS se conectan al nodo principal y se comparten con los nodos de cómputo a través de NFS. Esta opción puede ser rentable, pero el rendimiento depende de los recursos del nodo principal a medida que aumentan las necesidades de almacenamiento. Esto puede convertirse en un obstáculo a medida que se agreguen más nodos de cómputo al clúster y aumente la demanda de rendimiento.

Los sistemas de archivos Amazon EFS se escalan a medida que cambian las necesidades de almacenamiento. Puede configurar estos sistemas de archivos para una variedad de casos de uso. Utilice los sistemas de archivos Amazon EFS para ejecutar aplicaciones paralelizadas y sensibles a la latencia en su clúster.

Los sistemas de archivos FSx for Lustre pueden procesar conjuntos de datos masivos con un rendimiento de hasta cientos de gigabytes por segundo, millones de IOPS y latencias inferiores a milisegundos. Utilice los sistemas de archivos FSx for Lustre para entornos informáticos exigentes de alto rendimiento.

En el [Sección de SharedStorage](#), puede definir el almacenamiento externo o AWS ParallelCluster gestionado:

- El almacenamiento externo se refiere a un volumen o sistema de archivos existente que usted administra. AWS ParallelCluster no crea ni elimina este almacenamiento.
- El almacenamiento gestionado se refiere a un volumen o sistema de archivos que AWS ParallelCluster se creó y se puede eliminar.

Almacenamiento externo

Puede configurarlo AWS ParallelCluster para adjuntar almacenamiento externo al clúster cuando se cree o actualice el clúster. Del mismo modo, puede configurarlo para separar el almacenamiento externo del clúster cuando el clúster se elimine o actualice. Sus datos se conservan y puede utilizarlos para un almacenamiento compartido permanente a largo plazo fuera del ciclo de vida del clúster.

Note

Las versiones AWS ParallelCluster anteriores a la 3.8 no permiten montar sistemas de archivos gestionados externamente en `/home`. A partir de la versión 3.8, se AWS ParallelCluster puede utilizar `/home` como punto de montaje para un sistema de archivos gestionado externo. Puede montar un sistema de archivos gestionado externamente `/home` especificando `/home` el valor del [MountDir](#) parámetro en. [Sección de SharedStorage](#)

Amazon File Cache no es adecuado para su uso como /home directorio del sistema y, por lo tanto, no se admite su montaje en este momento/home.

Al especificar un /home directorio en [Sección de SharedStorage](#) la opción de [SharedStorageType](#) configuración, se anulará, lo que significa que se [Sección de SharedStorage](#) utilizará en su lugar la configuración de abajo.

Al montar un sistema de archivos externo en el /home directorio, AWS ParallelCluster copia el /home contenido del nodo principal al sistema de archivos externo, sin sobrescribir los archivos existentes en el almacenamiento externo. Esto incluye la transferencia de la clave SSH del clúster para el usuario predeterminado, si no está en el sistema de archivos externo.

Para obtener más información, consulte. [AWS ParallelCluster consideraciones sobre el almacenamiento compartido](#)

AWS ParallelCluster almacenamiento gestionado

AWS ParallelCluster el almacenamiento gestionado depende del ciclo de vida del clúster de forma predeterminada en la configuración. El parámetro de configuración `DeletionPolicy SharedStorage` se establece en `Delete` de forma predeterminada.

De forma predeterminada, un volumen o sistema de archivos AWS ParallelCluster gestionado y sus datos se eliminan si se cumple alguna de las siguientes condiciones.

- Ha eliminado el clúster
- Cambia la configuración del almacenamiento compartido gestionado `Name`.
- El almacenamiento compartido gestionado se elimina de la configuración.

Retain Configúrelo `DeletionPolicy` para conservar el volumen y los datos del sistema de archivos compartidos gestionados. Le recomendamos que haga copias de seguridad de sus datos con regularidad para evitar la pérdida de datos. Puede utilizarlo [AWS Backup](#) para gestionar de forma centralizada las copias de seguridad de todas sus opciones de almacenamiento.

Puede eliminar la dependencia del ciclo de vida mediante los ajustes de configuración. Para más información, consulte [Convierta el almacenamiento AWS ParallelCluster gestionado en almacenamiento externo](#).

Para obtener más información, consulte [Cuotas de almacenamiento compartido](#).

Para obtener más información sobre el almacenamiento compartido y el cambio a nuevas AWS ParallelCluster versiones, consulte [Mejores prácticas: mover un clúster a una nueva versión AWS ParallelCluster secundaria o a una versión de parche](#).

Puede configurarlo AWS ParallelCluster para adjuntar almacenamiento externo al clúster cuando se cree o actualice el clúster. Del mismo modo, puede configurarlo para separar el almacenamiento externo del clúster cuando el clúster se elimine o actualice. Sus datos se conservan y puede utilizarlos para soluciones de almacenamiento compartido permanente a largo plazo que son independientes del ciclo de vida del clúster.

De forma predeterminada, el almacenamiento gestionado depende del ciclo de vida del clúster. Puede eliminar la dependencia con los ajustes de configuración que se describen en [Convierta el almacenamiento AWS ParallelCluster gestionado en almacenamiento externo](#).

Con una configuración específica, puede optimizar cada una de las soluciones de almacenamiento compatibles para sus casos de uso.

Para conocer las cuotas de almacenamiento compartido, consulte [Cuotas de almacenamiento compartido](#).

Para obtener más información sobre el almacenamiento compartido y el cambio a nuevas AWS ParallelCluster versiones, consulte [Mejores prácticas: mover un clúster a una nueva versión AWS ParallelCluster secundaria o a una versión de parche](#).

Temas

- [Configuración del almacenamiento compartido](#)
- [Trabajar con almacenamiento compartido en AWS ParallelCluster](#)
- [Cuotas de almacenamiento compartido](#)

Configuración del almacenamiento compartido

Conozca los ajustes de configuración que puede utilizar para definir el almacenamiento compartido de su clúster.

Temas

- [Amazon Elastic Block Store](#)
- [Amazon Elastic File System](#)
- [Amazon FSx para Lustre](#)

- [Configuración de FSx para ONTAP, FSx para OpenZFS y almacenamiento compartido de caché de archivos](#)

Amazon Elastic Block Store

Si desea utilizar un volumen Amazon EBS existente para un almacenamiento permanente a largo plazo que sea independiente del ciclo de vida del clúster, especifique [EbsSettings](#) / [VolumeId](#).

Si no lo especifica [VolumeId](#), de forma predeterminada, AWS ParallelCluster crea un volumen de EBS gestionado a partir del [EbsSettings](#) momento en que se crea el clúster. AWS ParallelCluster también elimina el volumen y los datos cuando se elimina el clúster o se elimina el volumen de la configuración del clúster.

En el caso de un volumen de EBS AWS ParallelCluster gestionado, puede utilizar [EbsSettings/DeletionPolicy](#) para dar instrucciones AWS ParallelCluster DeleteRetain, o Snapshot el volumen cuando se elimine el clúster o cuando se elimine el volumen de la configuración del clúster. De forma predeterminada, DeletionPolicy está establecido en Delete.

Warning

En el AWS ParallelCluster caso del almacenamiento compartido gestionado, DeletionPolicy está configurado en de forma Delete predeterminada. Esto significa que, si se cumple una de las siguientes condiciones, se eliminan un volumen gestionado y sus datos:

- Ha eliminado el clúster
- Cambia la configuración del almacenamiento compartido gestionado [SharedStorage/Name](#).
- Quita el almacenamiento compartido gestionado de la configuración.

Le recomendamos que haga copias de seguridad del almacenamiento compartido con instantáneas de forma periódica para evitar la pérdida de datos. Para obtener más información sobre las instantáneas Amazon EBS, consulte [Instantáneas de Amazon EBS](#) en la Guía del usuario de instancias de Linux de Elastic Compute Cloud. Para obtener información sobre cómo administrar las copias de seguridad de datos en todas partes Servicios de AWS, consulte [AWSBackup](#) en la Guía para AWS Backup desarrolladores.

Amazon Elastic File System

Para utilizar un sistema de archivos Amazon EFS externo existente para el almacenamiento permanente a largo plazo fuera del ciclo de vida del clúster, especifique [EfsSettings/FileSystemId](#), de forma predeterminada, AWS ParallelCluster crea un sistema de archivos Amazon EFS gestionado desde el [EfsSettings](#) momento en que se crea el clúster. AWS ParallelCluster también elimina el sistema de archivos y los datos cuando se elimina el clúster o cuando se elimina el sistema de archivos de la configuración del clúster.

En el Retain caso de un sistema de archivos Amazon EFS AWS ParallelCluster gestionado, puede utilizar [EfsSettings/DeletionPolicy](#) para Delet e indicar cuándo se elimina el clúster o cuándo se elimina el sistema de archivos de la configuración del clúster. De forma predeterminada, DeletionPolicy está establecido en Delete.

Warning

En el caso del almacenamiento compartido AWS ParallelCluster gestionado, DeletionPolicy se establece en de forma predeterminada. Delete Esto significa que, si se cumple una de las siguientes condiciones, se eliminan un sistema de archivos gestionado y sus datos:

- Ha eliminado el clúster
- Cambia la configuración del almacenamiento compartido gestionado [SharedStorage/Name](#).
- Quita el almacenamiento compartido gestionado de la configuración.

Le recomendamos realizar una copia de seguridad de su almacenamiento compartido con regularidad para evitar la pérdida de datos. Para obtener más información sobre cómo realizar copias de seguridad de volúmenes individuales de Amazon EFS, consulte [Cómo hacer copias de seguridad de los sistemas de archivos de Amazon EFS](#) en la Guía del usuario de Amazon Elastic File System. Para obtener información sobre cómo administrar las copias de seguridad de datos en todas partes Servicios de AWS, consulte [AWSBackup](#) en la Guía para AWS Backup desarrolladores.

Amazon FSx para Lustre

Para utilizar un sistema de archivos FSx for Lustre externo existente para el almacenamiento permanente a largo plazo fuera del ciclo [FsxLustreSettings](#) de vida del clúster, especifique [FileSystemId](#).

Si no especifica [FsxLustreSettings/FileSystemId](#), de forma predeterminada, AWS ParallelCluster crea un [FsxLustreSettings](#) sistema de archivos FSx for Lustre administrado desde el momento en que crea el clúster. AWS ParallelCluster también elimina el sistema de archivos y los datos cuando se elimina el clúster o cuando el sistema de archivos se elimina de la configuración del clúster.

En el caso de un sistema de archivos FSx for Lustre AWS ParallelCluster gestionado, puede [FsxLustreSettings](#) utilizar [DeletionPolicy](#)/AWS ParallelCluster para Delete indicar Retain al sistema de archivos cuándo se elimina el clúster o cuando se elimina el sistema de archivos de la configuración del clúster. De forma predeterminada, [DeletionPolicy](#) está establecido en Delete.

Warning

En el caso del almacenamiento compartido AWS ParallelCluster gestionado, [DeletionPolicy](#) se establece en de forma predeterminada. Delete. Esto significa que, si se cumple una de las siguientes condiciones, se eliminan un sistema de archivos gestionado y sus datos:

- Ha eliminado el clúster
- Cambia la configuración del almacenamiento compartido gestionado [SharedStorage/Name](#).
- Quita el almacenamiento compartido gestionado de la configuración.

Le recomendamos realizar una copia de seguridad de su almacenamiento compartido con regularidad para evitar la pérdida de datos. Puede definir las copias de seguridad en su clúster con [SharedStorage/FsxLustreSettings/AutomaticBackupRetentionDays](#) [DailyAutomaticBackupStartTime](#). Para obtener información sobre cómo administrar las copias de seguridad de datos en todas partes Servicios de AWS, consulte [AWSBackup](#) en la Guía para AWS Backup desarrolladores.

Configuración de FSx para ONTAP, FSx para OpenZFS y almacenamiento compartido de caché de archivos

Para FSx para ONTAP, FSx para OpenZFS y File Cache, puede usar [FsxOntapSettings/VolumeId](#), [FsxOpenZfsSettings/y/](#) para especificar el montaje de un volumen externo existente o una caché de archivos [FileCacheId](#) para su clúster. [VolumeIdFileCacheSettings](#)

AWS ParallelCluster el almacenamiento compartido gestionado no es compatible con FSx para ONTAP, FSx para OpenZFS y File Cache.

Trabajar con almacenamiento compartido en AWS ParallelCluster

Obtén información sobre cómo trabajar con AWS ParallelCluster el almacenamiento compartido.

Temas

- [AWS ParallelCluster consideraciones sobre el almacenamiento compartido](#)
- [Convierta el almacenamiento AWS ParallelCluster gestionado en almacenamiento externo](#)

AWS ParallelCluster consideraciones sobre el almacenamiento compartido

Tenga en cuenta lo siguiente cuando trabaje con almacenamiento compartido en AWS ParallelCluster.

- Realice copias de seguridad de los datos del sistema de archivos con [AWS Backup](#) u otro método para gestionar las copias de seguridad de todos los sistemas de almacenamiento.
- Para añadir almacenamiento compartido, añada una sección de almacenamiento compartido al archivo de configuración y cree o actualice el clúster.
- Para eliminar el almacenamiento compartido, debe eliminar la sección de almacenamiento compartido del archivo de configuración y actualizar el clúster.
- Para sustituir el almacenamiento compartido AWS ParallelCluster gestionado existente por un nuevo almacenamiento gestionado, cambie el valor de [SharedStorage/Name](#) y actualice el clúster.

Warning

De forma predeterminada, el almacenamiento y los datos AWS ParallelCluster gestionados existentes se eliminan al realizar la actualización del clúster con un Name parámetro

nuevo. Si necesita cambiar Name y conservar los datos de almacenamiento compartido gestionado existentes, asegúrese de configurarlos Retain o de DeletionPolicy hacer una copia de seguridad de los datos antes de actualizar el clúster.

- Si no haces una copia de seguridad de los datos de almacenamiento AWS ParallelCluster gestionado y DeletionPolicy sí lo hacesDelete, tus datos se eliminarán cuando se elimine el clúster o cuando se elimine el almacenamiento gestionado de la configuración del clúster y el clúster se actualice.
- Si no haces una copia de seguridad de los datos de almacenamiento AWS ParallelCluster gestionado y DeletionPolicy sí lo hacesRetain, tu sistema de archivos se desconecta antes de eliminar el clúster y se puede volver a conectar a otro clúster como un sistema de archivos externo. Sus datos se conservan.
- Si el almacenamiento AWS ParallelCluster gestionado se elimina de la configuración del clúster y lo DeletionPolicy estáRetain, se puede volver a conectar al clúster como un sistema de archivos externo conservando los datos del clúster.
- A partir de AWS ParallelCluster la versión 3.4.0, puede mejorar la seguridad de los montajes del sistema de archivos de Amazon EFS configurando [SharedStorage/EfsSettings/EncryptionInTransity IamAuthorization](#)ajustes.
- Al montar un sistema de archivos externo en el directorio /home, AWS ParallelCluster copia el contenido del directorio /home del nodo principal al sistema de archivos externo. Copia los datos existentes en el directorio /home sin sobrescribir los archivos o directorios existentes en el almacenamiento externo. Esto incluye la clave SSH del clúster para el usuario predeterminado en caso de que aún no exista en el sistema de archivos externo. Por lo tanto, todos los demás clústeres que monten el mismo sistema de archivos externo en su directorio /home respectivo también tendrán la misma clave SSH para el usuario predeterminado del clúster.
- En un entorno de varios clústeres que monta el mismo sistema de archivos externo en los directorios /home de los clústeres, las claves SSH que permiten el acceso a los nodos de cómputo creadas en el nodo principal solo se generan una vez cuando el primer clúster monta el sistema de archivos externo en /home. AWS ParallelCluster Todos los demás clústeres utilizan la misma clave SSH. Como resultado, cualquier persona que posea la clave SSH del usuario predeterminado de estos clústeres compartidos puede acceder a cualquier clúster. Todos los nodos de cómputo permiten las conexiones mediante la clave generada inicialmente.

Convierta el almacenamiento AWS ParallelCluster gestionado en almacenamiento externo

Aprenda a convertir el almacenamiento AWS ParallelCluster gestionado en almacenamiento externo.

Los procedimientos se basan en el siguiente fragmento de archivo de configuración de ejemplo.

```
...
- MountDir: /fsx
  Name: fsx
  StorageType: FsxLustre
  FsxLustreSettings:
    StorageCapacity: 1200
    DeletionPolicy: Delete
...
```

Convierta el almacenamiento AWS ParallelCluster gestionado en almacenamiento externo

1. Establezca la `DeletionPolicy` a `Retain` en el archivo de configuración del clúster.

```
...
- MountDir: /fsx
  Name: fsx
  StorageType: FsxLustre
  FsxLustreSettings:
    StorageCapacity: 1200
    DeletionPolicy: Retain
...
```

2. Para realizar el cambio `DeletionPolicy`, ejecute el siguiente comando:

```
pcluster update-cluster -n cluster-name -c cluster-config.yaml
```

3. Elimine la `SharedStorage` sección del archivo de configuración del clúster.

```
...
...
```

4. Para cambiar lo gestionado `SharedStorage` a externo `SharedStorage` y separarlo del clúster, ejecute el siguiente comando.

```
pcluster update-cluster -n cluster-name -c cluster-config.yaml
```

5. El almacenamiento compartido ahora es externo y está separado del clúster.
6. Para conectar su sistema de archivos externo al clúster original o a otro clúster, siga estos pasos.
 - a. Consiga el ID del sistema de archivos de FSx para Lustre

- i. Para usarlo, AWS CLI ejecute el siguiente comando y busque el sistema de archivos con un nombre que incluya el nombre del clúster original y anote el ID del sistema de archivos.

```
aws fsx describe-file-systems
```

- ii. Inicie sesión en AWS Management Console y vaya a <https://console.aws.amazon.com/fsx/>. En la lista de sistemas de archivos, busque el sistema de archivos con un nombre que incluya el nombre del clúster original y anote el ID del sistema de archivos.
 - b. Actualice las reglas del grupo de seguridad del sistema de archivos para proporcionar acceso a y desde el sistema de archivos y las subredes del clúster. Puede encontrar el nombre y el ID del grupo de seguridad del sistema de archivos en la consola de Amazon FSx.

Agregue reglas al grupo de seguridad del sistema de archivos que permitan el tráfico TCP entrante y saliente desde y hacia los rangos o prefijos de CIDR IP del nodo principal y del nodo de cómputo. Especifique los puertos TCP 988, 1021, 1022 y 1023 para el tráfico TCP entrante y saliente.

Para obtener más información, consulte

[SharedStorage/FsxLustreSettings/FileSystemIdy Creación, configuración y eliminación de grupos de seguridad para Amazon EC2](#) en la Guía del AWS Command Line Interface usuario de la versión 2.

- c. Añada la SharedStorage sección a la configuración del clúster.

```
...  
- MountDir: /fsx  
  Name: fsx-external  
  StorageType: FsxLustre  
  FsxLustreSettings:
```

```
FileSystemId: fs-02e5b4b4abd62d51c
```

```
...
```

- d. Para agregar el almacenamiento compartido externo al clúster, ejecute el siguiente comando.

```
pcluster update-cluster -n cluster-name -c cluster-config.yaml
```

Cuotas de almacenamiento compartido

Configure el clúster SharedStorage para montar el almacenamiento de archivos compartido existente y crear un nuevo almacenamiento de archivos compartido en función de las cuotas que se indican en la siguiente tabla.

Las cuotas de almacenamiento de archivos montadas para cada clúster

Tipo de almacenamiento compartido de archivos	AWS ParallelCluster almacenamiento gestionado	Almacenamiento externo	Total neto de la cuota
Amazon EBS	5	5	5
REDADA	1	0	1
Amazon EFS	1	20	21
Amazon FSx †	1 FSx para Lustre	20	21

Note

Esta tabla de cuotas se ha añadido en la AWS ParallelCluster versión 3.2.0.

† AWS ParallelCluster solo admite el montaje de los sistemas Amazon FSx existentes para NetApp ONTAP, Amazon FSx para OpenZFS y File Cache. No admite la creación de nuevos sistemas FSx para ONTAP, FSx para OpenZFS y File Cache.

Note

Si lo utiliza AWS Batch como planificador, FSx for Lustre solo está disponible en el nodo principal del clúster.

Las cachés de archivos no son compatibles con los programadores. AWS Batch

Recursos y etiquetado de AWS ParallelCluster

Con AWS ParallelCluster, puede crear etiquetas para rastrear y administrar sus recursos de AWS ParallelCluster. Usted define las etiquetas que quiere que AWS CloudFormation cree y propague a todos los recursos del clúster en [Sección de Tags](#) del archivo de configuración del clúster. También puede utilizar etiquetas que AWS ParallelCluster genera automáticamente para realizar un seguimiento y administrar los recursos.

Al crear un clúster, el clúster y sus recursos se etiquetan con las etiquetas del sistema AWS ParallelCluster y AWS definidas en esta sección.

AWS ParallelCluster aplica etiquetas a las instancias, los volúmenes y los recursos del clúster. Para identificar la pila de clústeres, AWS CloudFormation aplica etiquetas del sistema AWS a las instancias del clúster. Para identificar las plantillas de lanzamiento de EC2, EC2 aplica etiquetas del sistema a las instancias del clúster. Puede utilizar etiquetas para ver y administrar los recursos de AWS ParallelCluster.

No puede modificar etiquetas del sistema AWS. Para evitar que esto afecte a la funcionalidad de AWS ParallelCluster, no modifique las etiquetas de AWS ParallelCluster.

A continuación, se muestra un ejemplo de una etiqueta del sistema AWS para un recurso de AWS ParallelCluster. No puede modificarla.

```
"aws:cloudformation:stack-name"="clustername"
```

A continuación, se muestra un ejemplo de una etiqueta de AWS ParallelCluster aplicada a un recurso. No la modifique.

```
"parallelcluster:cluster-name"="clustername"
```

Puede ver estas etiquetas en la sección EC2 de la AWS Management Console.

Ver etiquetas

1. Navegue hasta la consola EC2 en <https://console.aws.amazon.com/ec2/>.
2. Para ver todas las etiquetas del clúster, elija Etiquetas en el panel de navegación.
3. Para ver las etiquetas de clúster por instancia, elija Instancias en el panel de navegación.
4. Seleccione una instancia de clúster.
5. Seleccione la pestaña Administrar etiquetas en los detalles de la instancia y vea las etiquetas.
6. Elija la pestaña Almacenamiento en los detalles de la instancia.
7. Seleccione el ID de volumen.
8. En Volúmenes, seleccione el volumen.
9. Seleccione la pestaña Etiquetas en los detalles del volumen y vea las etiquetas.

Etiquetas de instancia del nodo principal de AWS ParallelCluster

Clave	Valor de etiqueta
<code>parallelcluster:cluster-name</code>	<i>clustername</i>
Name	HeadNode
<code>aws:ec2launchtemplate:id</code>	<i>lt-1234567890abcdef0</i>
<code>aws:ec2launchtemplate:version</code>	<i>1</i>
<code>parallelcluster:node-type</code>	HeadNode
<code>aws:cloudformation:stack-name</code>	<i>clustername</i>
<code>aws:cloudformation:logical-id</code>	HeadNode
<code>aws:cloudformation:stack-id</code>	<i>arn:aws:cloudformation: region-id :ACCOUNTID :stack/clustername /1234abcd-12ab-12ab-12ab-1234567890abcdef0</i>
<code>parallelcluster:version</code>	<i>3.7.0</i>

Etiquetas de volumen raíz del nodo principal de AWS ParallelCluster

Clave de etiqueta	Valor de etiqueta
<code>parallelcluster:cluster-name</code>	<i>clustername</i>
<code>parallelcluster:node-type</code>	HeadNode
<code>parallelcluster:version</code>	<i>3.7.0</i>

Etiquetas de instancia de nodo de computación de AWS ParallelCluster

Clave	Valor de etiqueta
<code>parallelcluster:cluster-name</code>	<i>clustername</i>
<code>parallelcluster:compute-resource-name</code>	<i>compute-resource-name</i>
<code>aws:ec2launchtemplate:id</code>	<i>lt-1234567890abcdef0</i>
<code>aws:ec2launchtemplate:version</code>	<i>1</i>
<code>parallelcluster:node-type</code>	Compute
<code>parallelcluster:queue-name</code>	<i>queue-name</i>
<code>parallelcluster:version</code>	<i>3.7.0</i>

Etiquetas de volumen raíz del nodo de computación de AWS ParallelCluster

Clave de etiqueta	Valor de etiqueta
<code>parallelcluster:cluster-name</code>	<i>clustername</i>
<code>parallelcluster:compute-resource-name</code>	<i>compute-resource-name</i>
<code>parallelcluster:node-type</code>	Compute

Clave de etiqueta	Valor de etiqueta
<code>parallelcluster:queue-name</code>	<i>queue-name</i>
<code>parallelcluster:version</code>	<i>3.7.0</i>

Etiquetas de la interfaz de usuario de AWS ParallelCluster

Clave de etiqueta	Valor de etiqueta
<code>parallelcluster-ui</code>	true

Monitoreo de AWS ParallelCluster y registros

La supervisión es una parte importante del mantenimiento de la fiabilidad, la disponibilidad y el rendimiento de AWS ParallelCluster y de sus otras soluciones de AWS. AWS ofrece las siguientes herramientas de supervisión para vigilar AWS ParallelCluster, informar cuando algo no va bien y tomar medidas automáticamente cuando proceda:

- Amazon CloudWatch monitorea los recursos de AWS y las aplicaciones que ejecuta en AWS en tiempo real. 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 hacer que CloudWatch haga un seguimiento del uso de la CPU u otras métricas de las instancias de Amazon EC2 y lanzar nuevas instancias automáticamente cuando sea necesario. Para obtener más información, consulte la [Guía del usuario de Amazon CloudWatch](#).
- Registros de Amazon CloudWatch le permite monitorear, almacenar y tener acceso a los archivos de registro desde instancias de Amazon EC2, CloudTrail u otras fuentes. CloudWatch Logs puede monitorear información en los registros y enviarle una notificación cuando se llega a determinados umbrales. También se pueden archivar los datos de los registros en un almacenamiento de larga duración. Para obtener más información, consulte la [Guía del usuario de Amazon CloudWatch Logs](#).
- AWS CloudTrail captura las llamadas a la API y otros eventos relacionados que realiza la Cuenta de AWS o se realizan en nombre de esta. Además, entrega los archivos de registros a un bucket de Amazon S3 especificado. También pueden identificar qué usuarios y cuentas llamaron a AWS,

la dirección IP de origen de las llamadas y el momento en que se hicieron. Para obtener más información, consulte la [Guía del usuario de AWS CloudTrail](#).

- Amazon EventBridge: es un bus de eventos sin servidor que facilita la conexión de sus aplicaciones con datos de varios orígenes. EventBridge proporciona un flujo de datos en tiempo real desde sus propias aplicaciones, aplicaciones de software como servicio (SaaS) y servicios de AWS y, luego, dirige dichos 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 la [Guía del usuario de Amazon EventBridge](#).

Temas

- [Integración con Amazon CloudWatch Logs](#)
- [Panel de Amazon CloudWatch](#)
- [Alarmas de Amazon CloudWatch para las métricas del clúster](#)
- [Rotación de registros configurada de AWS ParallelCluster](#)
- [Registros de CLI de pcluster](#)
- [Registros de salida de la consola EC2](#)
- [Recuperar registros AWS ParallelCluster de interfaz de usuario y AWS ParallelCluster tiempo de ejecución](#)
- [Recuperación y conservación de registros](#)

Integración con Amazon CloudWatch Logs

Para obtener más información acerca de los Registros de CloudWatch, consulte la [Guía del usuario de los Registros de Amazon CloudWatch](#). Para configurar la integración de los Registros de CloudWatch, consulte la sección [Monitoring](#). Para obtener información sobre cómo añadir registros personalizados a la configuración de CloudWatch mediante `append-config`, consulte [Varios archivos de configuración del agente de CloudWatch](#) en la Guía del usuario de Amazon CloudWatch.

Registros del clúster de los Registros de Amazon CloudWatch

Se crea un grupo de registros para cada clúster con un nombre, `/aws/parallelcluster/cluster-name-<timestamp>` (por ejemplo, `/aws/parallelcluster/testCluster-202202050215`). Cada registro (o conjunto de registros si la ruta contiene un `*`) de cada nodo tiene un flujo de registro

denominado `{hostname}.{instance_id}.{logIdentifier}`. (Por ejemplo: `ip-172-31-10-46.i-02587cf29cc3048f3.nodewatcher`). Los datos de registro se envían a CloudWatch a través del [agente de CloudWatch](#), que se ejecuta como root en todas las instancias del clúster.

Se crea un panel de Amazon CloudWatch cuando se crea el clúster. Este panel le permite revisar los registros almacenados en los Registros de CloudWatch. Para obtener más información, consulte [Panel de Amazon CloudWatch](#).

Esta lista contiene el *logIdentifier* y la ruta de los flujos de registro disponibles para las plataformas, los programadores y los nodos.

Flujos de registro disponibles para plataformas, programadores y nodos

Plataformas	Programadores	Nodos	Flujos de registro
amazon centos redhat ubuntu	awsbatc slurm	HeadNc	dcv-authenticator: /var/log/parallelcluster/pc luster_dcv_authenticator.log dcv-ext-authenticator: /var/log/parallelcluster/pc luster_dcv_connect.log dcv-agent: /var/log/dcv/agent.*.log dcv-xsession: /var/log/dcv/dcv-xsession.*.log dcv-server: /var/log/dcv/server.log dcv-session-launcher: /var/log/dcv/sessionlauncher.log Xdcv: /var/log/dcv/Xdcv.*.log cfn-init: /var/log/cfn-init.log chef-client: /var/log/chef-client.log
amazon centos redhat	awsbatc slurm	Comput eet HeadNc	cloud-init: /var/log/cloud-init.log supervisord: /var/log/supervisord.log

Plataformas	Programas	Nodos	Flujos de registro
ubuntu			
amazon centos redhat ubuntu	slurm	Comput eet	cloud-init-output: /var/log/cloud-init-output.log comptemgtd: /var/log/parallelcluster/comptemgtd slurmd: /var/log/slurmd.log slurm_prolog_epilog: /var/log/parallelcluster/sl urm_prolog_epilog.log
amazon centos redhat ubuntu	slurm	HeadNo	sssd: /var/log/sssd/sssd.log sssd_domain_default: /var/log/sssd/sssd_default.log pam_ssh_key_generator: /var/log/parallelcluster/pa m_ssh_key_generator.log clusterstatusmgt: /var/log/parallelcluster/cl usterstatusmgt clustermgtd: /var/log/parallelcluster/clustermgtd compute_console_output: /var/log/parallelcluster/co mpute_console_output slurm_resume: /var/log/parallelcluster/slurm_resum e.log slurm_suspend: /var/log/parallelcluster/slurm_suspe nd.log slurmctld: /var/log/slurmctld.log slurm_fleet_status_manager: /var/log/parallelcluster/sl urm_fleet_status_manager.log

Plataformas	Programas	Nodos	Flujos de registro
amazon centos redhat	awsbatch slurm	Compute HeadNode	system-messages: /var/log/messages
ubuntu	awsbatch slurm	Compute HeadNode	syslog: /var/log/syslog

Los trabajos de los clústeres que utilizan AWS Batch almacenan la salida de los trabajos que han alcanzado un estado de RUNNING, SUCCEEDED o FAILED en los Registros de CloudWatch. El grupo de registros es /aws/batch/job, y el formato del nombre del flujo de registro es *jobDefinitionName/default/ecs_task_id*. De forma predeterminada, estos registros están configurados para no caducar, pero se puede modificar el periodo de retención. Para obtener más información, consulte [Cambiar la retención de datos de registro en Registros de CloudWatch](#) en la Guía de usuario de Registros de Amazon CloudWatch.

Registros de imágenes de compilación de los Registros de Amazon CloudWatch

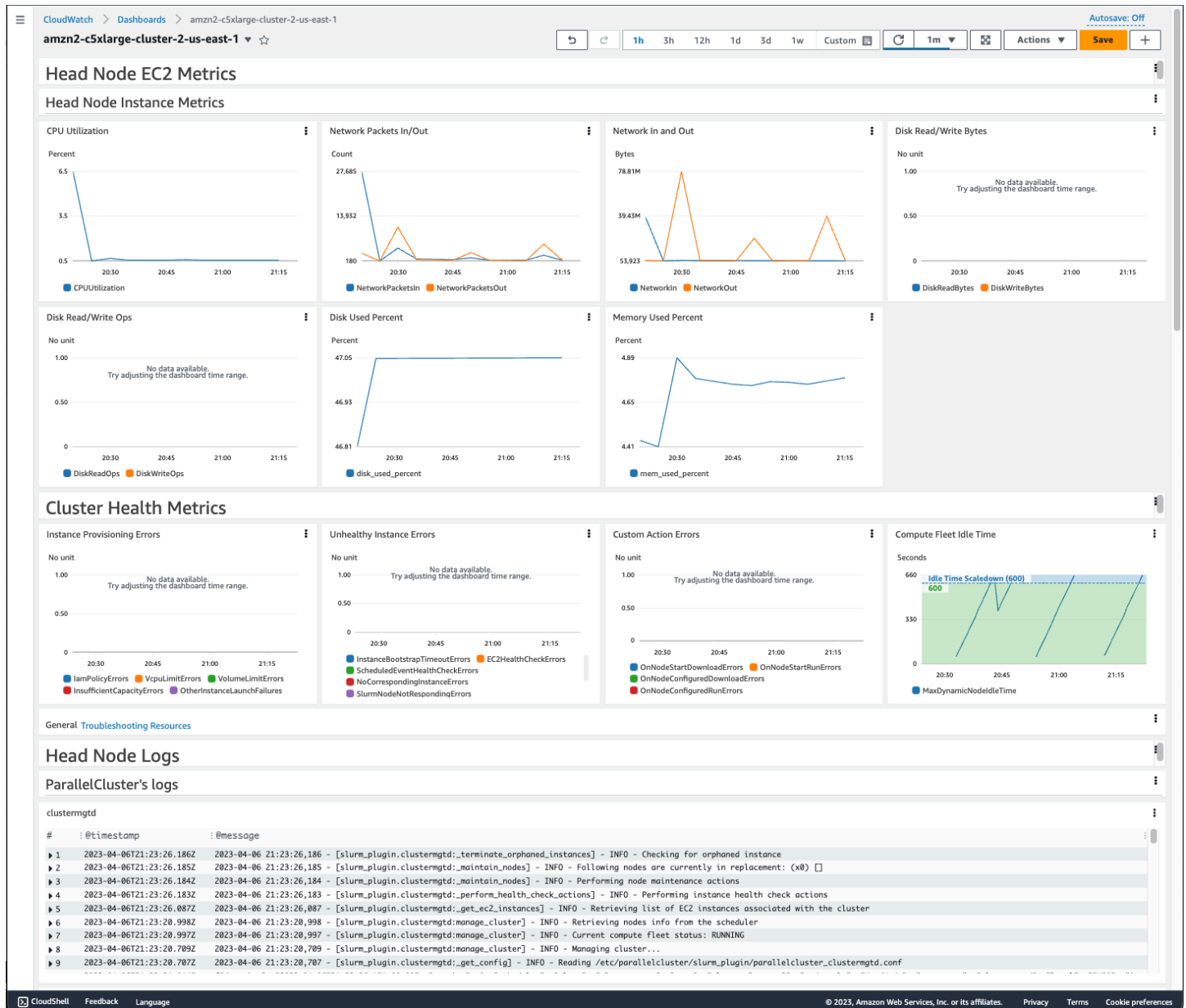
Se crea un grupo de registros para cada imagen de compilación personalizada con un nombre, /aws/imagebuilder/ParallelClusterImage-*<image-id>*. Un flujo de registro único con el nombre *{pcluster-version}/1* contiene la salida del proceso de imágenes de compilación.

Puede acceder a los registros mediante los comandos de imagen [pcluster](#). Para obtener más información, consulte [AWS ParallelCluster Personalización de AMI](#).

Panel de Amazon CloudWatch

Se crea un panel de Amazon CloudWatch cuando se crea un clúster. Esto facilita la supervisión de los nodos del clúster y la visualización de los registros almacenados en los Registros de Amazon CloudWatch. El nombre del panel es *ClusterName-Region*. *ClusterName* es el nombre del clúster y *Región* es la Región de AWS en la que se encuentra el clúster. Puede acceder al panel de control desde la consola o abriendo <https://console.aws.amazon.com/cloudwatch/home?region=Region#dashboards:name=ClusterName-Region>.

En la imagen siguiente, se muestra un ejemplo de un panel de CloudWatch para un clúster.



Métricas de la instancia del nodo principal

La primera sección del panel de control muestra gráficos de las métricas de EC2 del nodo principal.

Si su clúster tiene almacenamiento compartido, en la siguiente sección se muestran las métricas de almacenamiento compartido.

Métricas del estado del clúster

Si el clúster utiliza Slurm para la programación, los gráficos de métricas del estado del clúster muestran los errores de los nodos de computación del clúster en tiempo real. Para obtener más información, consulte [Solución de problemas de estado del clúster](#). Las métricas del estado del clúster se añaden al panel a partir de la versión 3.6.0 de AWS ParallelCluster.

Registros del nodo principal

En la sección final figuran los registros del nodo principal agrupados por registros de AWS ParallelCluster, registros del programador, registros de integración de NICE DCV y registros del sistema.

Para obtener información detallada acerca de los paneles de CloudWatch, consulte [Uso de los paneles de Amazon CloudWatch](#) en la Guía del usuario de Amazon CloudWatch.

Si no quiere crear el panel de Amazon CloudWatch, puede desactivarlo estableciendo [Monitoring/Dashboards/CloudWatch/Enabled](#) en false.

Note

Si deshabilita la creación del panel de Amazon CloudWatch, también se deshabilita Amazon CloudWatch y las alarmas `disk_used_percent` y `memory_used_percent` del clúster. Para obtener más información, consulte [Alarmas de Amazon CloudWatch para las métricas del clúster](#).

Las alarmas `disk_used_percent` y `memory_used_percent` se añaden a partir de la versión 3.6 de AWS ParallelCluster.

Alarmas de Amazon CloudWatch para las métricas del clúster

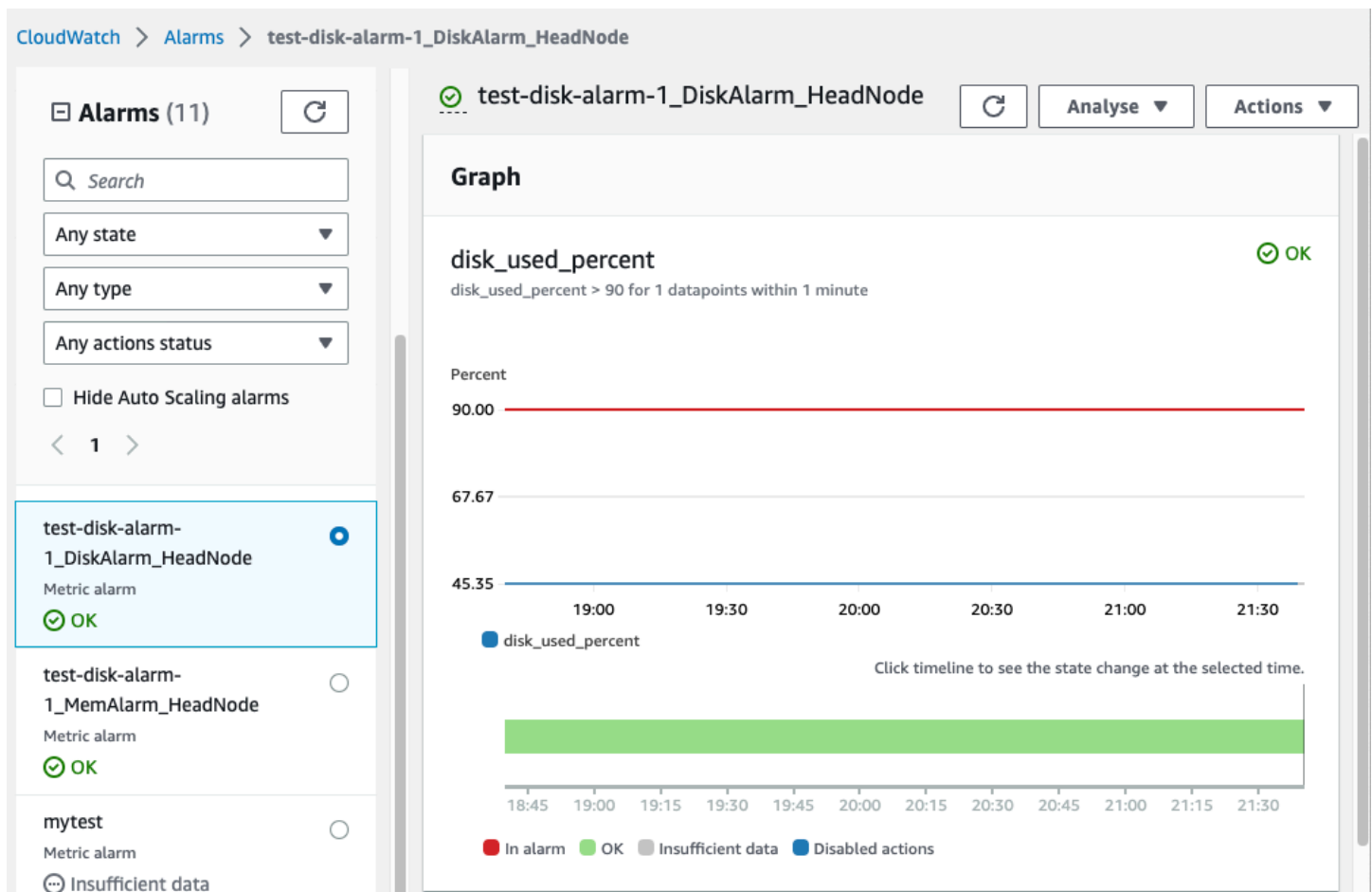
A partir de la versión 3.6 de AWS ParallelCluster, puede configurar el clúster con las alarmas de Amazon CloudWatch para monitorizar el nodo principal. Una alarma monitorea el volumen raíz `disk_used_percent`. La otra alarma monitorea la métrica `mem_used_percent`. Para obtener más información, consulte [Métricas recopiladas por el agente de CloudWatch](#) en la Guía del usuario de Amazon CloudWatch.

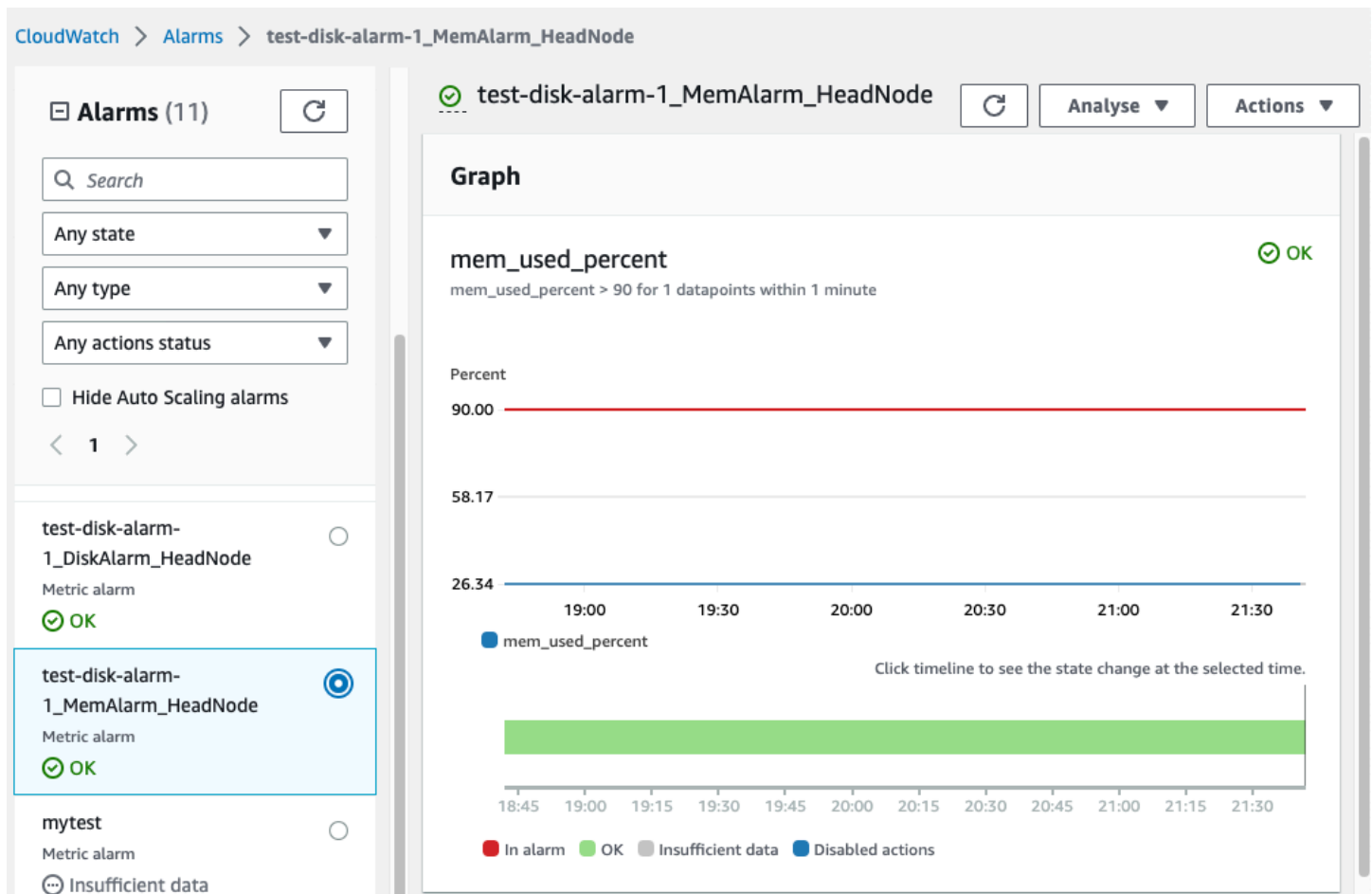
Las alarmas se denominan de la siguiente manera:

- `cluster-name_DiskAlarm_HeadNode`
- `cluster-name_MemAlarm_HeadNode`

cluster-name es el nombre del clúster.

Acceda a las alarmas de la consola CloudWatch seleccionando Alarmas en el panel de navegación. Las siguientes imágenes muestran la alarma de uso del disco y la alarma de uso de la memoria de un clúster.





La alarma de uso del disco se encuentra en estado ALARM cuando el porcentaje de uso del disco es superior al 90 % para 1 punto de datos, en un periodo de 1 minuto.

La alarma de uso de la memoria se encuentra en estado ALARM cuando el porcentaje de uso de la memoria es superior al 90 % para 1 punto de datos, en un periodo de 1 minuto.

Note

AWS ParallelCluster no configura las acciones de alarma de forma predeterminada. Para obtener información sobre cómo configurar las acciones de alarma, como el envío de notificaciones, consulte [Acciones de alarma](#). Para obtener información detallada acerca de las alarmas de CloudWatch, consulte [Uso de las alarmas de Amazon CloudWatch](#) en la Guía del usuario de Amazon CloudWatch.

Si no desea crear estas alarmas de Amazon CloudWatch, desactívelas estableciendo [Monitoring/Dashboards/CloudWatch/Enabled](#) como `false` en la configuración del clúster. Así

también se desactivará la creación del panel de control de Amazon CloudWatch. Para obtener más información, consulte [Panel de Amazon CloudWatch](#).

Note

Si desactiva la creación del panel de Amazon CloudWatch, también se desactiva Amazon CloudWatch y las alarmas `disk_used_percent` y `memory_used_percent` del clúster.

Rotación de registros configurada de AWS ParallelCluster

Las configuraciones de rotación de registros de AWS ParallelCluster se encuentran en archivos `/etc/logrotate.d/parallelcluster*_log_rotation`. Cuando un registro configurado rota, el contenido del registro actual se conserva en una sola copia de seguridad y el registro vacío reanuda la acción de registro.

Solo se mantiene una copia de seguridad por cada registro configurado.

AWS ParallelCluster configura un registro de crecimiento rápido para que rote cuando alcance un tamaño de 50 MB. Los registros de crecimiento rápido están relacionados con el escalado y Slurm, por ejemplo `/var/log/parallelcluster/clustermgtd`, `/var/log/parallelcluster/slurm_resume.log` y `/var/log/slurmctld.log`.

AWS ParallelCluster configura un registro de crecimiento lento para que rote cuando alcance un tamaño de 10 MB.

Puede ver los registros anteriores que se retienen durante el número de días definido en el ajuste [Logs/CloudWatch/RetentionInDays](#) de la configuración del clúster con el registro de CloudFormation activado. Compruebe la configuración de `RetentionInDays` para ver si es necesario aumentar el número de días según su caso de uso.

AWS ParallelCluster configura y rota los siguientes registros:

Registros del nodo principal

```
/var/log/cloud-init.log
/var/log/supervisord.log
/var/log/cfn-init.log
/var/log/chef-client.log
```

```
/var/log/dcv/server.log
/var/log/dcv/sessionlauncher.log
/var/log/dcv/agent.*.log
/var/log/dcv/dcv-xsession.*.log
/var/log/dcv/Xdcv.*.log
/var/log/parallelcluster/pam_ssh_key_generator.log
/var/log/parallelcluster/clustermgtd
/var/log/parallelcluster/clusterstatusmgtd
/var/log/parallelcluster/slurm_fleet_status_manager.log
/var/log/parallelcluster/slurm_resume.log
/var/log/parallelcluster/slurm_suspend.log
/var/log/slurmctld.log
/var/log/slurmdbd.log
/var/log/parallelcluster/compute_console_output.log
```

Registros de nodos de computación

```
/var/log/cloud-init.log
/var/log/supervisord.log
/var/log/cloud-init-output.log
/var/log/parallelcluster/computemgtd
/var/log/slurmd.log
```

Registros de nodos de inicio de sesión

```
/var/log/cloud-init.log
/var/log/cloud-init.log
/var/log/cloud-init-output.log
/var/log/supervisord.log
/var/log/parallelcluster/pam_ssh_key_generator.log
```

Registros de CLI de **pcluster**

La CLI de `pcluster` escribe los registros de sus comandos en archivos `pcluster.log.#` en `/home/user/.parallelcluster/`.

Para cada comando, los registros suelen incluir el comando con las entradas, una copia de la versión de la API de la CLI utilizada para crear el comando, la respuesta y los mensajes de información y error. En el caso de un comando de creación y compilación, los registros también incluyen el archivo de configuración, las operaciones de validación del archivo de configuración, la plantilla de CloudFormation y los comandos de pila.

Puede usar estos registros para verificar errores, entradas, versiones y comandos de la CLI de `pcluster`. También pueden servir como registro de cuándo se emitieron los comandos.

Registros de salida de la consola EC2

Cuando AWS ParallelCluster detecta que una instancia de nodo de computación estática finaliza de forma inesperada, intenta recuperar la salida de la consola EC2 de la instancia de nodo terminada una vez transcurrido un periodo de tiempo. De esta forma, si el nodo de computación no ha podido comunicarse con Amazon CloudWatch, es posible que se siga obteniendo de la salida de la consola información útil para la solución de problemas sobre por qué finalizó el nodo. Esta salida de la consola se registra en el registro `/var/log/parallelcluster/compute_console_output` del nodo principal. Para obtener más información acerca de la salida de EC2, consulte [Salida de la consola](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

De forma predeterminada, AWS ParallelCluster solo recupera la salida de la consola de un subconjunto de ejemplos de nodos finalizados. Esto evita que el nodo principal del clúster se vea abrumado por varias solicitudes de salida de la consola provocadas por un gran número de finalizaciones. De forma predeterminada, AWS ParallelCluster espera cinco minutos entre la detección de la finalización y la recuperación de la salida de la consola para que EC2 tenga tiempo de recuperar la salida final de la consola de los nodos.

Puede editar el tamaño de la muestra y los valores de los parámetros del tiempo de espera en el archivo `/etc/parallelcluster/slurm_plugin/parallelcluster_clustermgtd.conf` del nodo principal.

Esta característica se añade en la AWS ParallelCluster versión 3.5.0.

Parámetros de salida de la consola EC2

Puede editar los valores de los siguientes parámetros de salida de la consola EC2 en el archivo `/etc/parallelcluster/slurm_plugin/parallelcluster_clustermgtd.conf` del nodo principal.

`compute_console_logging_enabled`

Para deshabilitar la recopilación de registros de salida de la consola, establezca `compute_console_logging_enabled` en `false`. El valor predeterminado es `true`.

Puede actualizar este parámetro en cualquier momento, sin detener la flota de computación.

`compute_console_logging_max_sample_size`

`compute_console_logging_max_sample_size` establece el número máximo de nodos de computación desde los que AWS ParallelCluster recopila las salidas de la consola cada vez que detecta una finalización inesperada. Si este valor es inferior a 1, AWS ParallelCluster recupera la salida de la consola de todos los nodos finalizados. El valor predeterminado es 1.

Puede actualizar este parámetro en cualquier momento, sin detener la flota de computación.

`compute_console_wait_time`

`compute_console_wait_time` establece el tiempo, en segundos, que AWS ParallelCluster espera entre la detección de un fallo en un nodo y la recopilación de la salida de la consola desde ese nodo. Puede aumentar el tiempo de espera si determina que EC2 necesita más tiempo para recopilar la salida final del nodo finalizado. El valor predeterminado es de 300 segundos (5 minutos).

Puede actualizar este parámetro en cualquier momento, sin detener la flota de computación.

Recuperar registros AWS ParallelCluster de interfaz de usuario y AWS ParallelCluster tiempo de ejecución

Aprenda a recuperar la AWS ParallelCluster interfaz de usuario y los registros AWS ParallelCluster de tiempo de ejecución para solucionar problemas. Para empezar, busca la AWS ParallelCluster interfaz de usuario y los nombres de las AWS ParallelCluster pilas correspondientes. Utilice el nombre de la pila para localizar los grupos de registros de instalación. Para terminar, exporte los registros. Estos registros son específicos del AWS ParallelCluster tiempo de ejecución. Para los registros de clúster, consulte [Recuperación y conservación de registros](#).

Requisitos previos

- El AWS CLI está instalado.
- Tiene credenciales para ejecutar AWS CLI comandos en la AWS ParallelCluster interfaz de usuario en la Cuenta de AWS que se encuentra.
- Puede acceder a la consola Amazon CloudWatch en la Cuenta de AWS que esté activada la AWS ParallelCluster interfaz de usuario.

Paso 1: busque los nombres de las pilas correspondientes

En el siguiente ejemplo, reemplace el texto resaltado en rojo con sus valores reales.

Enumere las pilas utilizando el Región de AWS lugar donde instaló la AWS ParallelCluster interfaz de usuario:

```
$ aws cloudformation list-stacks --region aws-region-id
```

Anote los nombres de las pilas de las siguientes pilas:

- El nombre de la pila que implementó la AWS ParallelCluster interfaz de usuario en tu cuenta. Introdujo este nombre al instalar la AWS ParallelCluster interfaz de usuario; por ejemplo, `pcluster-ui`.
- La AWS ParallelCluster pila que lleva el prefijo del nombre de pila que ha introducido; por ejemplo, `pcluster-ui-ParallelClusterApi-ABCD1234EFGH`.

Paso 2: Localice los grupos de registros

Listar los grupos de registros de la pila de AWS ParallelCluster UI, tal y como se muestra en el siguiente ejemplo: siguiente:

```
$ aws cloudformation describe-stack-resources \  
  --region aws-region-id \  
  --stack-name pcluster-ui \  
  --query "StackResources[?ResourceType == 'AWS::Logs::LogGroup' &&  
(LogicalResourceId == 'ApiGatewayAccessLog' || LogicalResourceId ==  
'ParallelClusterUILambdaLogGroup')].PhysicalResourceId" \  
  --output text
```

Listar los grupos de registros de la pila de AWS ParallelCluster API, tal y como se muestra en el siguiente ejemplo:

```
$ aws cloudformation describe-stack-resources \  
  --region aws-region-id \  
  --stack-name pcluster-ui-ParallelCluster-Api-ABCD1234EFGH \  
  --query "StackResources[?ResourceType == 'AWS::Logs::LogGroup' && LogicalResourceId  
== 'ParallelClusterFunctionLogGroup'].PhysicalResourceId" \  
  --output text
```

observar las listas de grupos de registros que se utilizarán en el siguiente paso.

Paso 3: exportar los registros

Siga los siguientes pasos para recopilar y exportar los registros:

1. Inicie sesión en la AWS Management Console consola [Amazon CloudWatch](#) en la que se encuentre la Cuenta de AWS AWS ParallelCluster interfaz de usuario y navegue hasta ella.
2. En el panel de navegación, elija Registros y luego, Logs Insights en el panel de navegación.
3. Seleccione todos los grupos de registros que se muestran en el paso anterior.
4. Elija un intervalo de tiempo, como 12 horas.
5. Ejecute la siguiente consulta:

```
$ fields @timestamp, @message
| sort @timestamp desc
| limit 10000
```

6. Selecciona Exportar resultados, descargar tabla (JSON).

Recuperación y conservación de registros

AWS ParallelCluster crea métricas de EC2 para las instancias y el almacenamiento de HeadNode y Compute. Puede ver las métricas en los paneles personalizados de la consola CloudWatch. AWS ParallelCluster también crea flujos de registro de CloudWatch del clúster en grupos de registros. Puede ver estos registros en los paneles personalizados o grupos de registros de la consola CloudWatch. En la sección de configuración del clúster [Monitoreo](#), se describe cómo se pueden modificar los registros y el panel de CloudWatch del clúster. Para obtener más información, consulte [Integración con Amazon CloudWatch Logs](#) y [Panel de Amazon CloudWatch](#).

Los registros son un recurso útil para solucionar problemas. Por ejemplo, si desea eliminar un clúster que falla, puede ser útil crear primero un archivo con los registros del clúster. Siga los pasos que se indican en [Registros de archivos](#) para crear un archivo.

Temas

- [Registros del clúster no disponibles en CloudWatch](#)
- [Registros de archivos](#)
- [Registros conservados](#)
- [Registros de nodos finalizados](#)

Registros del clúster no disponibles en CloudWatch

Si los registros del clúster no están disponibles en CloudWatch, asegúrese de no haber sobrescrito la configuración de registros de CloudWatch de AWS ParallelCluster al añadir registros personalizados a la configuración.

Para añadir registros personalizados a la configuración de CloudWatch, asegúrese de añadirlos a la configuración en lugar de buscarlos y sobrescribirlos. Para obtener más información sobre `fetch-config` y `append-config`, consulte [Varios archivos de configuración del agente de CloudWatch](#) en la Guía del usuario de CloudWatch.

Para restaurar la configuración de registro de CloudWatch de AWS ParallelCluster, puede ejecutar los siguientes comandos dentro de un nodo de AWS ParallelCluster:

```
$ PLATFORM="$(ohai platform | jq -r ".[]")"
LOG_GROUP_NAME="$(cat /etc/chef/dna.json | jq -r ".cluster.log_group_name")"
SCHEDULER="$(cat /etc/chef/dna.json | jq -r ".cluster.scheduler")"
NODE_ROLE="$(cat /etc/chef/dna.json | jq -r ".cluster.node_type")"
CONFIG_DATA_PATH="/usr/local/etc/cloudwatch_agent_config.json"
/opt/parallelcluster/pyenv/versions/cookbook_virtualenv/bin/python /usr/local/bin/
write_cloudwatch_agent_json.py --platform $PLATFORM --config $CONFIG_DATA_PATH --log-
group $LOG_GROUP_NAME --scheduler $SCHEDULER --node-role $NODE_ROLE
/opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2
-c file:/opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.json -s
```

Registros de archivos

Puede archivar los registros en Amazon S3 o en un archivo local (según el parámetro `--output-file`).

Note

Añada permisos a la política del bucket de Amazon S3 para conceder acceso a CloudWatch. Para obtener más información, consulte [Establecer permisos en un bucket de Amazon S3](#) en la Guía del usuario de los Registros de CloudWatch.

```
$ pcluster export-cluster-logs --cluster-name mycluster --region eu-west-1 \
  --bucket bucketname --bucket-prefix logs
{
```

```
"url": "https://bucketname.s3.eu-west-1.amazonaws.com/export-log/mycluster-logs-202109071136.tar.gz?..."
}

# use the --output-file parameter to save the logs locally
$ pcluster export-cluster-logs --cluster-name mycluster --region eu-west-1 \
  --bucket bucketname --bucket-prefix logs --output-file /tmp/archive.tar.gz
{
  "path": "/tmp/archive.tar.gz"
}
```

El archivo contiene los eventos de pila de AWS CloudFormation y de flujos de Registros de Amazon CloudWatch desde el nodo principal y los nodos de computación de los últimos 14 días, a menos que se especifique explícitamente en la configuración o en los parámetros del comando `export-cluster-logs`. El tiempo que tarda el comando en finalizar depende del número de nodos del clúster y del número de flujos de registro disponibles en los Registros de CloudWatch. Para obtener más información sobre las opciones disponibles, consulte [Integración con Amazon CloudWatch Logs](#).

Registros conservados

A partir de la versión 3.0.0, AWS ParallelCluster conserva los Registros de CloudWatch de forma predeterminada cuando se elimina un clúster. Si quiere eliminar un clúster y conservar sus registros, asegúrese de que [Monitoring/Logs/CloudWatch/DeletionPolicy](#) no esté establecido como Delete en la configuración del clúster. De lo contrario, cambie el valor de este campo a Retain y ejecute el comando `pcluster update-cluster`. A continuación, ejecute `pcluster delete-cluster --cluster-name <cluster_name>` para eliminar el clúster, pero retenga el grupo de registros que está almacenado en Amazon CloudWatch.

Registros de nodos finalizados

Si un nodo de computación estático finaliza inesperadamente y CloudWatch no tiene registros, compruebe si AWS ParallelCluster ha registrado la salida de la consola para ese nodo de computación en el nodo principal del registro `/var/log/parallelcluster/compute_console_output`. Para obtener más información, consulte [Registros clave para la depuración](#).

Si el registro `/var/log/parallelcluster/compute_console_output` no está disponible o no contiene la salida del nodo, utilice la AWS CLI para recuperar la salida de la consola del nodo que ha fallado. Inicie sesión en el nodo principal del clúster y obtenga el nodo fallido `instance-id` del archivo `/var/log/parallelcluster/slurm_resume.log`.

Recupere la salida de la consola mediante el siguiente comando con el `instance-id`:

```
$ aws ec2 get-console-output --instance-id i-abcdef01234567890
```

Si un nodo de computación dinámico finaliza automáticamente tras su lanzamiento y CloudWatch no tiene registros de este, envíe un trabajo que active una acción de escalado de clústeres. Espere a que se produzca un error en la instancia y recupere el registro de la consola de la instancia.

Inicie sesión en el nodo principal del clúster y obtenga el nodo de computación `instance-id` del archivo `/var/log/parallelcluster/slurm_resume.log`.

Recupere el registro de la consola de la instancia mediante el siguiente comando:

```
$ aws ec2 get-console-output --instance-id i-abcdef01234567890
```

El registro de salidas de la consola puede ayudarle a depurar la causa raíz de un error en un nodo de computación cuando el registro del nodo de computación no esté disponible.

AWS CloudFormation recurso personalizado

A partir de AWS ParallelCluster la versión 3.6.0, puede usar un recurso AWS ParallelCluster CloudFormation personalizado en una AWS CloudFormation pila. El recurso personalizado es una pila AWS ParallelCluster alojada. De esta forma, puede usarlo CloudFormation para configurar y administrar sus clústeres. Por ejemplo, puede configurar los recursos externos del clúster, como la red, el almacenamiento compartido y la infraestructura de grupos de seguridad en una CloudFormation pila. Además, puede administrar su clúster con una CloudFormation infraestructura como canalización de código.

Añada un recurso AWS ParallelCluster personalizado a su CloudFormation plantilla de la siguiente manera:

1. Agrega una pila de proveedores de recursos personalizada que sea propiedad de y esté alojada por AWS ParallelCluster.
2. Haz referencia a la pila de proveedores CloudFormation de tu plantilla como un recurso personalizado.

La pila de proveedores de recursos personalizada gestiona las CloudFormation solicitudes y responde a ellas. Por ejemplo, cuando despliegas tu CloudFormation pila, también configuras y creas

un clúster. Para actualizar un clúster, debes actualizar tu CloudFormation pila. Al eliminar la pila, se elimina un clúster. Para obtener más información sobre los recursos CloudFormation personalizados, consulta [los recursos personalizados](#) en la Guía del AWS CloudFormation usuario.

Warning

CloudFormation no detecta la desviación personalizada de los recursos. Úselo únicamente CloudFormation para actualizar la configuración del clúster y eliminar un clúster.

Puede usar la CLI de [pcluster](#) o [IU AWS ParallelCluster](#) para monitorear el estado del clúster o para actualizar la flota de computación, pero no para actualizar la configuración del clúster ni para eliminarlo.

Note

Le recomendamos que añada una [protección de finalización](#) a su pila para evitar eliminarla accidentalmente.

Pila de proveedores alojada por AWS ParallelCluster

La pila de proveedores de recursos personalizada tiene el formato que se muestra en el siguiente fragmento CloudFormation de plantilla:

```
PclusterClusterProvider:
  Type: AWS::CloudFormation::Stack
  Properties:
    Parameters:
      CustomLambdaRole: # (Optional) RoleARN to override default
      AdditionalIamPolicies: # (Optional) comma-separated list of IAM policies to add
    TemplateURL: !Sub
      - https://${AWS::Region}-aws-parallelcluster.s3.${AWS::Region}.${AWS::URLSuffix}/
parallelcluster/${Version}/templates/custom_resource/cluster.yaml
      - { Version: 3.7.0 }
```

Propiedades:

Parámetros:

CustomLambdaRole (opcional):

Un rol personalizado con permisos para ejecutar el AWS Lambda que crea y administra el clúster. De forma predeterminada, el rol usa las mismas políticas definidas de forma predeterminada en la [documentación de AWS ParallelCluster](#).

AdditionalIamPolicies (opcional):

Una lista separada por comas de nombres de recursos de Amazon (ARN) adicionales de la política de IAM para añadirlos al rol que utiliza Lambda. Solo se usa si no se especifica un CustomLambdaRole y se puede dejar en blanco.

Si necesita políticas adicionales para el nodo principal, los nodos de computación o para acceder a un bucket de Amazon S3, agréguelas a la propiedad CustomLambdaRole o AdditionalIamPolicy.

Para obtener más información sobre las políticas predeterminadas, consulte [AWS Identity and Access Management permisos en AWS ParallelCluster](#).

TemplateURL (obligatorio):

La URL del archivo de recursos AWS ParallelCluster personalizado.

Salidas:

ServiceToken:

Un valor que se puede utilizar como propiedad de recursos personalizados de ServiceToken. Un recurso personalizado ServiceToken especifica dónde se AWS CloudFormation envían las solicitudes. Se trata de una entrada obligatoria para un recurso de clúster que incluya en la AWS CloudFormation plantilla.

LogGroupArn:

El ARN en el CloudWatch LogGroup que se registra el recurso subyacente.

LambdaLayerArn:

El ARN de la capa Lambda que se utiliza para ejecutar las operaciones. AWS ParallelCluster

Recurso del clúster

El recurso del CloudFormation clúster tiene el formato que se muestra en el siguiente fragmento de plantilla: CloudFormation

```
PclusterCluster:
  Type: Custom::PclusterCluster
  Properties:
    ServiceToken: !GetAtt [ PclusterClusterProvider , Outputs.ServiceToken ]
    ClusterName: !Sub 'c-${AWS::StackName}' # Must be different from StackName
    ClusterConfiguration:
      # Your Cluster Configuration
```

Propiedades:

ServiceToken:

El resultado de la pila AWS ParallelCluster de proveedores. ServiceToken

ClusterName:

El nombre del clúster que se va a crear y administrar. El nombre no debe coincidir con el nombre de la CloudFormation pila. El nombre no se puede cambiar una vez creado el clúster.

ClusterConfiguration:

El archivo YAML de configuración del clúster, tal como se describe en [Configuración del clúster](#). Sin embargo, puede utilizar las CloudFormation construcciones habituales, como las [funciones intrínsecas](#).

DeletionPolicy:

Define si se debe eliminar el clúster cuando se elimina la pila raíz. El valor predeterminado es Delete.

Retain:

Retiene el clúster si se elimina el recurso personalizado.

Note

Para que el clúster retenido siga funcionando, los recursos que dependen del clúster, como el almacenamiento y las redes, deben tener establecida una política de eliminación en retener.

Delete:

Elimina el clúster si se elimina el recurso personalizado.

Valores devueltos de Fn::GetAtt:

La función intrínseca de Fn::GetAtt devuelve un valor para un atributo especificado de un tipo. Para obtener más información sobre el uso de la Fn::GetAtt intrínseca función, consulte [Fn::GetAtt](#)

ClusterProperties:

Los valores de la operación [pcluster describe-cluster](#).

validationMessages:

Una cadena que contiene todos los mensajes de validación que se produjeron durante la última operación de creación o actualización.

logGroupName:

El nombre del grupo de registros que se utiliza para registrar las operaciones del clúster de Lambda. Los eventos de registro se retienen durante 90 días y el grupo de registros se retiene después de eliminar el clúster.

Ejemplo: Fn::GetAtt:

```
# Provide the public IP address of the head node as an output of a stack
Outputs:
  HeadNodeIp:
    Description: The public IP address of the head node
    Value: !GetAtt [ PclusterCluster, headNode.publicIpAddress ]
```

Ejemplo: CloudFormation plantilla sencilla y completa con un recurso AWS ParallelCluster personalizado:

```
AWSTemplateFormatVersion: '2010-09-09'
Description: > AWS ParallelCluster CloudFormation Template

Parameters:
  HeadNodeSubnet:
    Description: Subnet where the HeadNode will run
    Type: AWS::EC2::Subnet::Id
```

```
ComputeSubnet:
  Description: Subnet where the Compute Nodes will run
  Type: AWS::EC2::Subnet::Id

KeyName:
  Description: KeyPair to login to the head node
  Type: AWS::EC2::KeyPair::KeyName

Resources:
  PclusterClusterProvider:
    Type: AWS::CloudFormation::Stack
    Properties:
      TemplateURL: !Sub
        - https://${AWS::Region}-aws-parallelcluster.s3.${AWS::Region}.
          ${AWS::URLSuffix}/parallelcluster/${Version}/templates/custom_resource/cluster.yaml
        - { Version: 3.7.0 }

  PclusterCluster:
    Type: Custom::PclusterCluster
    Properties:
      ServiceToken: !GetAtt [ PclusterClusterProvider , Outputs.ServiceToken ]
      ClusterName: !Sub 'c-${AWS::StackName}'
      ClusterConfiguration:
        Image:
          Os: alinux2
        HeadNode:
          InstanceType: t2.medium
          Networking:
            SubnetId: !Ref HeadNodeSubnet
          Ssh:
            KeyName: !Ref KeyName
        Scheduling:
          Scheduler: slurm
          SlurmQueues:
            - Name: queue0
              ComputeResources:
                - Name: queue0-cr0
                  InstanceType: t2.micro
              Networking:
                SubnetIds:
                  - !Ref ComputeSubnet

Outputs:
```



```
HeadNodeIp:
  Description: The Public IP address of the HeadNode
  Value: !GetAtt [ PclusterCluster, headNode.publicIpAddress ]
ValidationMessages:
  Description: Any warnings from cluster create or update operations.
  Value: !GetAtt PclusterCluster.validationMessages
```

Para obtener más información sobre cómo utilizar el recurso CloudFormation AWS ParallelCluster personalizado, consulte [Crear un clúster con AWS CloudFormation](#).

Operaciones de clúster

Cuando se agrega un recurso personalizado de clúster a una CloudFormation pila, CloudFormation puede realizar las siguientes operaciones de clúster:

- CloudFormation crea un clúster en una pila nueva e independiente cuando despliega una pila que incluye el recurso AWS ParallelCluster personalizado.
- Si actualiza la configuración del clúster definida en la pila, de acuerdo con las políticas de actualización de la configuración, CloudFormation actualiza el clúster. El proveedor de recursos AWS ParallelCluster personalizados no detiene la flota de procesamiento antes de actualizar el clúster. Le recomendamos que utilice la configuración [QueueUpdateStrategy](#) con las actualizaciones del clúster. De esta forma, puedes evitar realizar `pcluster update-compute-fleet` llamadas explícitas antes y después de las actualizaciones cuando utilices el recurso AWS ParallelCluster personalizado.
- Si elimina la pila, se elimina el clúster.

Pilas de solución de problemas que incluyen el recurso AWS ParallelCluster personalizado

Con un recurso AWS ParallelCluster personalizado, CloudFormation implementa un clúster desde una pila nueva e independiente. Puede supervisar la creación de clústeres siguiendo estos pasos:

1. Navega hasta CloudFormation Stacks AWS Management Console y selecciona Stacks en el panel de navegación.
2. Elija la pila con el nombre que definió para el nombre del clúster.
3. Si el estado de la pila es `ROLLBACK_COMPLETE`, se ha producido un error durante la creación del clúster.

4. Seleccione Detalles de la pila y, a continuación, la pestaña Eventos.
5. Busque el nombre que definió para el nombre del clúster en Eventos en ID lógico. Tiene un `Status reason` que indica el motivo del problema.
6. También puede elegir el menú desplegable Pilas y luego Eliminado para ver la lista de pilas eliminadas. Seleccione la pila con el nombre del clúster y consulte Eventos para obtener más información.
7. Para ver el resultado del proveedor de recursos personalizado que administra el clúster, selecciona la pila con la descripción «Recurso personalizado del AWS ParallelCluster clúster». Seleccione la pestaña Recursos, busque el recurso con el ID lógico `PcClusterCfnFunctionLogGroup` y siga el enlace indicado. Vea los flujos de registro que muestran la salida de la depuración de Lambda.
8. Para solucionar los problemas del clúster, consulte [AWS ParallelCluster solución de problemas](#).

Elastic Fabric Adapter

Elastic Fabric Adapter (EFA) es un dispositivo de red que tiene funciones de omisión del sistema operativo para las comunicaciones en red de baja latencia con otras instancias en la misma subred. EFA se expone mediante Libfabric, y lo pueden utilizar las aplicaciones que utilizan Messaging Passing Interface (MPI).

Para usar EFA con AWS ParallelCluster un Slurm programador, defina [SlurmQueues//ComputeResourcesEfa/Enabled](#) en `en. true`

Para ver la lista de instancias de EC2 compatibles con EFA, consulte [Tipos de instancias admitidas](#) en la Guía del usuario para instancias de Linux de Amazon EC2.

Le recomendamos que ejecute las instancias habilitadas para EFA en un grupo de ubicación. De esta forma, las instancias se lanzan en un grupo de baja latencia en una única zona de disponibilidad. Para obtener más información sobre cómo configurar los grupos de ubicación con AWS ParallelCluster, consulte [SlurmQueues/Networking/PlacementGroup](#).

Para obtener más información, consulte [Elastic Fabric Adapter](#) en la Guía del usuario de Amazon EC2 y [escale las cargas de trabajo de HPC con el adaptador Elastic Fabric y AWS ParallelCluster](#) en el AWS blog de código abierto.

Note

Elastic Fabric Adapter (EFA) no es compatible con diferentes zonas de disponibilidad. [Para obtener más información, consulte `Scheduling//Networking SlurmQueues/. SubnetIds`](#)

Note

De forma predeterminada, las distribuciones de Ubuntu habilitan la protección (seguimiento del proceso) de `ptrace`. La protección de `ptrace` está deshabilitada para que `Libfabric` funcione correctamente. Para obtener más información, consulte [Desactivar la protección `ptrace`](#) en la Guía del usuario de Amazon EC2.

Habilitar Intel MPI

Intel MPI está disponible en las AMI de AWS ParallelCluster para los valores `alinux2`, `centos7`, `rhe18`, `ubuntu2204` y `ubuntu2004` del ajuste [Image/Os](#).

Note

Para utilizar Intel MPI, debe reconocer y aceptar los términos de la [licencia de software simplificada de Intel](#).

De forma predeterminada, Open MPI se instala en la ruta. Para habilitar Intel MPI en lugar de Open MPI, debe cargar primero el módulo de Intel MPI. A continuación, debe instalar la versión más reciente utilizando `module load intelmpi`. El nombre exacto del módulo cambia con cada actualización. Para saber qué módulos están disponibles, ejecute `module avail`. El resultado es el siguiente.

```
$ module avail
-----/usr/share/Modules/modulefiles
-----
dot                modules
libfabric-aws/1.16.0~amzn3.0  null
module-git        openmpi/4.1.4
module-info       use.own
```

```
-----/opt/intel/mpi/2021.6.0/modulefiles
-----
intelmpi
```

Para cargar un módulo, ejecute `module load modulename`. Puede añadir esto al script utilizado para ejecutar `mpirun`.

```
$ module load intelmpi
```

Para saber qué módulos están cargados, ejecute `module list`.

```
$ module list
Currently Loaded Modulefiles:
 1) intelmpi
```

Para comprobar que Intel MPI está habilitado, ejecute `mpirun --version`.

```
$ mpirun --version
Intel(R) MPI Library for Linux* OS, Version 2021.6 Build 20220227 (id: 28877f3f32)
Copyright 2003-2022, Intel Corporation.
```

Una vez cargado el módulo Intel MPI, se cambian varias rutas para utilizar las herramientas de Intel MPI. Para ejecutar código compilado por las herramientas de Intel MPI, cargue primero el módulo Intel MPI.

Note

Intel MPI no es compatible con las instancias basadas en Graviton de AWS.

Note

Antes de la versión 2.5.0 de AWS ParallelCluster, Intel MPI no estaba disponible en las AMI de AWS ParallelCluster en las regiones China (Pekín) y China (Ningxia).

API de AWS ParallelCluster

¿Qué es la API de AWS ParallelCluster?

La API de AWS ParallelCluster es una aplicación sin servidor que, una vez implementada en su Cuenta de AWS, proporciona acceso programático a las características de AWS ParallelCluster a través de una API.

La API de AWS ParallelCluster se distribuye como una plantilla [AWS CloudFormation](#) independiente que incluye un punto de conexión de [Amazon API Gateway](#), que expone las características de AWS ParallelCluster, y una función de [AWS Lambda](#), que se encarga de procesar las características invocadas.

En la siguiente imagen se muestra un diagrama de alto nivel de la infraestructura de la API de AWS ParallelCluster.

Documentación de la API de AWS ParallelCluster

El archivo de especificaciones de OpenAPI que describe la API de AWS ParallelCluster se puede descargar desde:

```
https://<REGION>-aws-parallelcluster.s3.<REGION>.amazonaws.com/  
parallelcluster/<VERSION>/api/ParallelCluster.openapi.yaml
```

A partir del archivo de especificaciones de OpenAPI, puede generar documentación para la API de AWS ParallelCluster utilizando una de las muchas herramientas disponibles, como [Swagger UI](#) o [Redoc](#).

Cómo implementar la API de AWS ParallelCluster

Para implementar la API de AWS ParallelCluster, debe ser administrador de Cuenta de AWS.

La plantilla utilizada para implementar la API está disponible en la siguiente URL:

```
https://<REGION>-aws-parallelcluster.s3.<REGION>.amazonaws.com/  
parallelcluster/<VERSION>/api/parallelcluster-api.yaml
```

donde **<REGION>** es la Región de AWS en la que se debe implementar la API y donde **<VERSION>** es la versión de AWS ParallelCluster (por ejemplo, 3.7.0).

AWS Lambda procesa las características invocadas por la API mediante una interfaz de capa de Lambda con [API de la biblioteca Python de AWS ParallelCluster](#).

⚠ Warning

Cualquier usuario de la Cuenta de AWS, que tenga acceso con privilegios a los servicios de Amazon API Gateway o AWS Lambda, hereda automáticamente los permisos para administrar los recursos de la API de AWS ParallelCluster.

Implementación con AWS CLI

Configure las credenciales de AWS para usarlas con la CLI si aún no lo ha hecho.

```
$ aws configure
```

Ejecute los siguientes comandos para implementar la API:

```
$ REGION=<region>
$ API_STACK_NAME=<stack-name> # This can be any name
$ VERSION=3.7.0
$ aws cloudformation create-stack \
  --region ${REGION} \
  --stack-name ${API_STACK_NAME} \
  --template-url https://${REGION}-aws-parallelcluster.s3.${REGION}.amazonaws.com/
parallelcluster/${VERSION}/api/parallelcluster-api.yaml \
  --capabilities CAPABILITY_NAMED_IAM CAPABILITY_AUTO_EXPAND
$ aws cloudformation wait stack-create-complete --stack-name ${API_STACK_NAME} --region
${REGION}
```


Personalización de la implementación

Puede personalizar la implementación de la API mediante los parámetros AWS CloudFormation expuestos en la plantilla. Para configurar el valor de un parámetro al implementar a través de la CLI, se puede usar la siguiente opción: `--parameters ParameterKey=KeyName,ParameterValue=Value`.

Los siguientes parámetros son opcionales:

- **Región:** utilice el parámetro `Region` para especificar si la API puede controlar recursos en todas las Regiones de AWS (opción predeterminada) o en solo una Región de AWS. Establezca este valor en la Región de AWS en la que se esté implementando la API para restringir el acceso.


- `ParallelClusterFunctionRole`- Esto anula la función de IAM que se asigna a la AWS Lambda función que implementa AWS ParallelCluster las funciones. El parámetro acepta el ARN de un rol de IAM. Este rol debe configurarse para que AWS Lambda sea la entidad principal de IAM.
- `CustomDomainName`, `CustomDomainCertificate`, `CustomDomainHostedZoneId` - Utilice estos parámetros para configurar un dominio personalizado para el punto final de Amazon API Gateway. `CustomDomainName` es el nombre del dominio que se va a utilizar, `CustomDomainCertificate` es el ARN de un certificado AWS administrado para este nombre de dominio y `CustomDomainHostedZoneId` es el ID de la zona alojada de [Amazon Route 53](#) en la que desea crear registros.

 Warning

Puede configurar ajustes de dominio personalizados para aplicar una versión mínima de seguridad de la capa de transporte (TLS) para la API. Para obtener más información, consulte [Elección de una versión mínima de TLS para un dominio personalizado en API Gateway](#).

- `EnableIamAdminAccess`- De forma predeterminada, la AWS Lambda función que procesa las operaciones de la AWS ParallelCluster API está configurada con una función de IAM que impide cualquier acceso de IAM privilegiado (). `EnableIamAdminAccess=false` Esto hace que la API no pueda procesar operaciones que requieren la creación de políticas o roles de IAM. Por este motivo, la creación de clústeres o imágenes personalizadas solo se realiza correctamente cuando se proporcionan los roles de IAM como entrada en la configuración de los recursos.

Cuando `EnableIamAdminAccess` se establece en `true`, la API de AWS ParallelCluster concede permisos para administrar la creación de los roles de IAM necesarios para implementar clústeres o generar AMI personalizadas.

 Warning

Si se establece en verdadero, se otorgan privilegios de administrador de IAM a las operaciones de AWS ParallelCluster de procesamiento de la función de AWS Lambda.

Consulte [AWS ParallelCluster ejemplos de políticas de usuario para administrar los recursos de IAM](#) para obtener más información sobre las características que se pueden desbloquear al habilitar este modo.

- **PermissionsBoundaryPolicy**- Este parámetro opcional acepta un ARN de política de IAM existente que se establecerá como límite de permisos para todas las funciones de IAM creadas por la infraestructura de la API de PC y como condición de los permisos administrativos de IAM, de modo que la API de PC solo pueda crear funciones con esta política.

Consulte [Modo PermissionsBoundary](#) para obtener más información sobre las restricciones que impone este modo.

- **CreateApiUserRole**- De forma predeterminada, el despliegue de la AWS ParallelCluster API incluye la creación de una función de IAM que se establece como la única función autorizada para invocar la API. El punto de enlace Amazon API Gateway está configurado con una política basada en recursos para conceder el permiso de invocación únicamente al usuario creado. Para cambiarlo, defina `CreateApiUserRole=false` y, a continuación, conceda el acceso a la API a los usuarios de IAM seleccionados. Para obtener más información, consulte [Control del acceso para invocar una API](#) en la Guía para desarrolladores de Amazon API Gateway.

Warning

Cuando el acceso de `CreateApiUserRole=true` al punto de conexión de la API no está restringido por las políticas de recursos de Amazon API Gateway, todos los roles de IAM que tengan permisos de `execute-api:Invoke` ilimitados pueden acceder a las características de AWS ParallelCluster. Para obtener más información, consulte [Control del acceso a una API con las políticas de recursos de API Gateway](#) en la Guía para desarrolladores de API Gateway.

Warning

El `ParallelClusterApiUserRole` tiene permiso para invocar todas las operaciones de la API de AWS ParallelCluster. Para restringir el acceso a un subconjunto de recursos de la API, consulte la sección [Control de quién puede llamar a un método de API de API Gateway con políticas de IAM](#) en la Guía para desarrolladores de API Gateway.

- **IAM RoleAndPolicyPrefix**: este parámetro opcional acepta una cadena de 10 caracteres como máximo que se utilizará como prefijo tanto para las funciones como para las políticas de IAM creadas como parte de la infraestructura de API de PC.

Actualización de la API

Actualización a una versión más reciente de AWS ParallelCluster

Opción 1: elimine la API existente eliminando la pila de AWS CloudFormation correspondiente e implementando la nueva API, tal y como se muestra arriba.

Opción 2: actualice la API existente mediante la ejecución de los siguientes comandos:

```
$ REGION=<region>
$ API_STACK_NAME=<stack-name> # This needs to correspond to the existing API stack
name
$ VERSION=3.7.0
$ aws cloudformation update-stack \
  --region ${REGION} \
  --stack-name ${API_STACK_NAME} \
  --template-url https://${REGION}-aws-parallelcluster.s3.${REGION}.amazonaws.com/
parallelcluster/${VERSION}/api/parallelcluster-api.yaml \
  --capabilities CAPABILITY_NAMED_IAM CAPABILITY_AUTO_EXPAND
$ aws cloudformation wait stack-update-complete --stack-name ${API_STACK_NAME} --region
${REGION}
```

Invocación de la API de AWS ParallelCluster

El punto de conexión de Amazon API Gateway de AWS ParallelCluster está configurado con un [tipo de autorización de AWS_IAM](#) y requiere que todas las solicitudes estén firmadas mediante SigV4 con credenciales de IAM válidas ([referencia de API: realizar solicitudes http](#)).

Cuando se implementa con la configuración predeterminada, los permisos de invocación de la API solo se otorgan al usuario de IAM predeterminado creado con la API.

Para recuperar el ARN del usuario de IAM predeterminado, ejecute:

```
$ REGION=<region>
$ API_STACK_NAME=<stack-name>
$ aws cloudformation describe-stacks --region ${REGION} --stack-name ${API_STACK_NAME}
  --query "Stacks[0].Outputs[?OutputKey=='ParallelClusterApiUserRole'].OutputValue" --
output text
```

[Para obtener credenciales temporales para el usuario de IAM predeterminado, ejecute el comando STS. AssumeRole](#)

Puede obtener el punto de conexión de la API de AWS ParallelCluster ejecutando el siguiente comando:

```
$ REGION=<region>
$ API_STACK_NAME=<stack-name>
$ aws cloudformation describe-stacks --region ${REGION} --stack-name ${API_STACK_NAME}
  --query "Stacks[0].Outputs[?OutputKey=='ParallelClusterApiInvokeUrl'].OutputValue" --
output text
```

Cualquier cliente HTTP que cumpla con las especificaciones de OpenAPI que se encuentran a continuación puede invocar la API de AWS ParallelCluster:

```
https://<REGION>-aws-parallelcluster.s3.<REGION>.amazonaws.com/
parallelcluster/<VERSION>/api/ParallelCluster.openapi.yaml
```

Las solicitudes deben estar firmadas mediante SigV4, tal y como se indica [aquí](#).

En este momento, no ofrecemos ninguna implementación de cliente de API oficial. Sin embargo, los clientes de API se pueden generar fácilmente a partir del modelo OpenAPI mediante [OpenAPI Generator](#). Una vez generado el cliente, es necesario añadir la firma SigV4 si no se proporciona lista para usar.

Puede encontrar una implementación de referencia para un cliente de API de Python en el [repositorio de AWS ParallelCluster](#). Para obtener más información sobre cómo puede utilizar el cliente de API de Python, consulte el tutorial [Uso de la API AWS ParallelCluster](#).

Para implementar mecanismos de control de acceso más avanzados, como Amazon Cognito o Lambda Authorizers, o para proteger aún más la API con claves de API o AWS WAF, consulte la [documentación de Amazon API Gateway](#).

Warning

Un usuario de IAM que esté autorizado a invocar la API de AWS ParallelCluster puede controlar indirectamente todos los recursos de AWS administrados por AWS ParallelCluster en la Cuenta de AWS. Esto incluye la creación de recursos de AWS que el usuario no puede controlar directamente debido a las restricciones de la política de IAM del usuario. Por ejemplo, la creación de un clúster de AWS ParallelCluster, según su configuración, puede incluir la implementación de instancias de Amazon EC2, Amazon Route 53, sistemas de archivos Amazon Elastic File System, sistemas de archivos Amazon FSx, roles de IAM y

recursos de otros Servicios de AWS utilizados por AWS ParallelCluster sobre los que el usuario podría no tener control directo.

Warning

Al crear un clúster con las `AdditionalIamPolicies` especificadas en la configuración, las políticas adicionales deben coincidir con uno de los siguientes patrones:

```
- !Sub arn:${AWS::Partition}:iam:${AWS::AccountId}:policy/parallelcluster*
- !Sub arn:${AWS::Partition}:iam:${AWS::AccountId}:policy/parallelcluster/*
- !Sub arn:${AWS::Partition}:iam:aws:policy/CloudWatchAgentServerPolicy
- !Sub arn:${AWS::Partition}:iam:aws:policy/AmazonSSMManagedInstanceCore
- !Sub arn:${AWS::Partition}:iam:aws:policy/AWSBatchFullAccess
- !Sub arn:${AWS::Partition}:iam:aws:policy/AmazonS3ReadOnlyAccess
- !Sub arn:${AWS::Partition}:iam:aws:policy/service-role/AWSBatchServiceRole
- !Sub arn:${AWS::Partition}:iam:aws:policy/service-role/
  AmazonEC2ContainerServiceforEC2Role
- !Sub arn:${AWS::Partition}:iam:aws:policy/service-role/
  AmazonECSTaskExecutionRolePolicy
- !Sub arn:${AWS::Partition}:iam:aws:policy/service-role/
  AmazonEC2SpotFleetTaggingRole
- !Sub arn:${AWS::Partition}:iam:aws:policy/EC2InstanceProfileForImageBuilder
- !Sub arn:${AWS::Partition}:iam:aws:policy/service-role/
  AWSLambdaBasicExecutionRole
```

Si necesita otras políticas adicionales, puede elegir una de las siguientes opciones:

- Edite la `DefaultParallelClusterIamAdminPolicy` en:

```
https://<REGION>-aws-parallelcluster.s3.<REGION>.amazonaws.com/
parallelcluster/<VERSION>/api/parallelcluster-api.yaml
```

Agregue la política en la sección `ArnLike/iam:PolicyARN`.

- Omite especificar políticas para `AdditionalIamPolicies` en el archivo de configuración y añada políticas manualmente al rol de instancia de AWS ParallelCluster creado en el clúster.

Acceso a registros y métricas de API

Los registros de API se publican en Amazon CloudWatch con una retención de 30 días. Para recuperar el LogGroup nombre asociado a una implementación de API, ejecuta el siguiente comando:

```
$ REGION=<region>
$ API_STACK_NAME=<stack-name>
$ aws cloudformation describe-stacks --region ${REGION} --
stack-name ${API_STACK_NAME} --query "Stacks[0].Outputs[?
OutputKey=='ParallelClusterLambdaLogGroup'].OutputValue" --output text
```

También se puede acceder a las métricas, los registros y los registros rastreo [AWS X-Ray](#) de Lambda a través de la consola Lambda. Para recuperar el ARN de la función de Lambda asociada a una implementación de API, ejecute el siguiente comando:

```
$ REGION=<region>
$ API_STACK_NAME=<stack-name>
$ aws cloudformation describe-stacks --region ${REGION} --stack-name ${API_STACK_NAME}
--query "Stacks[0].Outputs[?OutputKey=='ParallelClusterLambdaArn'].OutputValue" --
output text
```

Conexión al nodo principal a través de NICE DCV

NICE DCV es una tecnología de visualización remota que permite a los usuarios conectarse de forma segura a aplicaciones 3D con gráficos avanzados que se alojan en un servidor remoto de alto rendimiento. Para obtener más información, consulte [NICE DCV](#).

El software NICE DCV se instala automáticamente en el nodo principal y se puede habilitar mediante la sección [Dcv](#) de la configuración del [HeadNode](#).

```
HeadNode:
  Dcv:
    Enabled: true
```

De esta forma, AWS ParallelCluster establece `/home/<DEFAULT_AMI_USER>` del nodo principal en la [carpeta de almacenamiento del servidor DCV](#). Para obtener más información acerca de los parámetros de configuración de NICE DCV, consulte [HeadNode/Dcv](#). Para conectarse a la sesión de NICE DCV, utilice el comando [pcluster dcv-connect](#).

Certificado HTTPS NICE DCV

NICE DCV genera automáticamente un certificado autofirmado para proteger el tráfico entre el cliente de NICE DCV y el servidor NICE DCV.

Para reemplazar el certificado autofirmado de NICE DCV predeterminado por otro certificado, conéctese primero al nodo principal. A continuación, copie el certificado y la clave en la carpeta /etc/dcv antes de ejecutar el comando [pcluster dcv-connect](#).

Para obtener más información, consulte [Cambio del certificado TLS](#) en la Guía del administrador de NICE DCV.

Licencias de NICE DCV

El servidor de NICE DCV no requiere un servidor de licencias cuando se ejecuta en instancias de Amazon EC2. Sin embargo, el servidor de NICE DCV debe conectarse periódicamente a un bucket de Amazon S3 para determinar si hay una licencia válida disponible.

AWS ParallelCluster agrega automáticamente los permisos necesarios a la política de IAM del nodo principal. Cuando utilice una política de instancias de IAM personalizada, use los permisos descritos en [NICE DCV en Amazon EC2](#) en la Guía del administrador de NICE DCV.

Para obtener sugerencias acerca de la solución de problemas, consulte [Solución de problemas de NICE DCV](#).

Uso de `pcluster update-cluster`

En la AWS ParallelCluster versión 3.x, [pcluster update-cluster](#) analiza la configuración utilizada para crear el clúster actual y la configuración del archivo de configuración para detectar problemas. Si se descubre algún problema, se informa al respecto y se muestran los pasos a seguir para solucionarlo. Por ejemplo, si el [InstanceType](#) de computación se cambia, se debe detener la flota de computación para que se pueda continuar con la actualización. Este problema se notifica cuando se descubre. Si no se detecta ningún problema de bloqueo, se inicia el proceso de actualización y se informa de los cambios.

Puede utilizar el [pcluster update-cluster --dryrun](#) option para ver los cambios antes de que se ejecuten. Para obtener más información, consulte [Ejemplos de pcluster update-cluster](#).

Para obtener ayuda sobre la solución de problemas, consulte [AWS ParallelCluster solución de problemas](#).

Política de actualización: definiciones

Política de actualización: esta configuración se puede cambiar durante una actualización.

Tras cambiar esta configuración, el clúster se puede actualizar mediante [pcluster update-cluster](#).

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Tras cambiar esta configuración, el clúster no se puede actualizar. Debe revertir la configuración del clúster original y crear uno nuevo con la configuración actualizada. Puede eliminar el clúster original más adelante. Utilice la [pcluster create-cluster](#) para crear un clúster. Para eliminar el clúster original, use [pcluster delete-cluster](#).

Política de actualización: esta configuración no se analiza durante una actualización.

Tras cambiar esta configuración, el clúster se puede actualizar mediante [pcluster update-cluster](#).

Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.

Estos ajustes no se pueden cambiar mientras exista la flota de computación. El cambio debe revertirse o la flota de computación debe detenerse (usando [pcluster update-compute-fleet](#)). Una vez detenida la flota de computación, puede actualizar el clúster ([pcluster update-cluster](#)) para activar los cambios. Por ejemplo, si utiliza un programador de Slurm con [SlurmQueues/ComputeResources/-Name/MinCount](#) > 0, se iniciará una flota de computación.

Política de actualización: la flota de cómputo y los nodos de inicio de sesión deben estar detenidos para poder cambiar esta configuración y poder actualizarse.

Esta configuración no se puede cambiar mientras exista la flota informática o si los nodos de inicio de sesión estén en uso. Se debe revertir el cambio o se deben detener la flota informática y los nodos de inicio de sesión (la flota informática se puede dejar de utilizar [pcluster update-compute-fleet](#)). Una vez que se hayan detenido la flota informática y los nodos de inicio de sesión, puede actualizar el clúster ([pcluster update-cluster](#)) para activar los cambios.

Política de actualización: esta configuración no se puede reducir durante una actualización.

Estos ajustes se pueden cambiar, pero no se pueden reducir. Debe revertir la configuración del clúster original y crear uno nuevo con la configuración actualizada. Puede eliminar el clúster


original más adelante. Utilice la [pcluster create-cluster](#) para crear un clúster. Para eliminar el clúster original, use [pcluster delete-cluster](#).

Política de actualización: si se cambia esta configuración, no se permite la actualización. Si fuerzas la actualización, se ignorará el nuevo valor y se utilizará el anterior.

Tras cambiar esta configuración, el clúster no se puede actualizar. Debe revertir la configuración del clúster original y crear uno nuevo con la configuración actualizada. Puede eliminar el clúster original más adelante. Utilice la [pcluster create-cluster](#) para crear un clúster. Para eliminar el clúster original, use [pcluster delete-cluster](#).

Política de actualización: debe haberse detenido la flota de computación o configurado [QueueUpdateStrategy](#) para poder cambiar esta configuración y proceder a la actualización.

Estos ajustes se pueden cambiar. La flota de computación debe estar detenida (utilizando [pcluster update-compute-fleet](#)) o [QueueUpdateStrategy](#) debe configurarse. Tras detener la flota de computación o configurar [QueueUpdateStrategy](#), puede actualizar el clúster ([pcluster update-cluster](#)) para activar los cambios.

 Note


Esta política de actualización se admite a partir de la AWS ParallelCluster versión 3.2.0.

Política de actualización: para esta configuración de valores de lista, se puede añadir un nuevo valor durante una actualización o se debe detener la flota de computación al eliminar un valor existente.

Se puede añadir un nuevo valor a estos ajustes durante una actualización. Tras añadir un nuevo valor a la lista, el clúster se puede actualizar mediante ([pcluster update-cluster](#)).

Para eliminar un valor existente de la lista, se debe detener la flota de computación (utilizando [pcluster update-compute-fleet](#)).

Por ejemplo, si usa un Slurm programador y agrega un nuevo tipo de [instancia a Instances/ InstanceType](#), puede actualizar el clúster sin detener la flota de cómputo. [Para eliminar un tipo de instancia existente de Instances/ InstanceType, primero se debe detener la flota de procesamiento \(mediante pcluster\). update-compute-fleet](#)

 Note

Esta política de actualización se admite a partir de la versión 3.2.0. AWS ParallelCluster

Política de actualización: para reducir el tamaño de una cola, es necesario detener la flota informática o [QueueUpdateStrategy](#) configurarla en TERMINATE para poder cambiar esta configuración para una actualización.

Esta configuración se puede cambiar, pero si el cambio quiere reducir el tamaño de la cola, la flota de procesamiento debe detenerse (mediante `pcluster update-compute-fleet`) o configurarse en [QueueUpdateStrategy](#) TERMINATE. Una vez que la flota de procesamiento se detenga o [QueueUpdateStrategy](#) se establezca en TERMINATE, puede actualizar el clúster (`pcluster update-cluster`) para activar los cambios.

El comando TERMINATE establecido al cambiar la capacidad del clúster solo terminará los nodos del final de la lista de nodos y dejará intactos todos los demás nodos de la misma partición.

Por ejemplo, si la capacidad inicial del clúster es `MinCount = 5` y `MaxCount = 10`, los nodos son. `st-[1-5]`; `dy-[1-5]` Al cambiar el tamaño del clúster a `MinCount = 3` y `MaxCount = 5`, la nueva capacidad del clúster estará compuesta por los nodos `st-[1-3]`; `dy-[1-2]`, que no se modificarán durante la actualización. Durante la actualización, solo se cerrarán los nodos `st-[4-5]`; `dy-[3-5]`.

Se admiten los siguientes cambios y no es necesario detener la flota informática ni [QueueUpdateStrategy](#) configurarla en TERMINATE:

- Se añade [SlurmQueue](#) uno nuevo
- Se añade [ComputeResource](#) una nueva
- [MaxCount](#) se incrementa
- [MinCount](#) se incrementa y [MaxCount](#) se incrementa al menos en la misma cantidad

Nota: Esta política de actualización se admite a partir de la AWS ParallelCluster versión 3.9.0.

Política de actualización: para configurar esta lista de valores, debe detenerse la flota de computación o configurarse [QueueUpdateStrategy](#) para añadir un nuevo valor; la flota de computación debe detenerse al eliminar un valor existente.

Se puede añadir un nuevo valor a estos ajustes durante una actualización. La flota de computación debe estar detenida (utilizando `pcluster update-compute-fleet`) o [QueueUpdateStrategy](#) debe configurarse. Tras detener la flota de computación o configurar [QueueUpdateStrategy](#), puede actualizar el clúster (`pcluster update-cluster`) para activar los cambios.

Para eliminar un valor existente de la lista, se debe detener la flota de computación (utilizando `pcluster update-compute-fleet`).

Note

Esta política de actualización se admite a partir de la AWS ParallelCluster versión 3.3.0.

Política de actualización: todos los nodos de procesamiento deben estar detenidos para poder eliminar un grupo de ubicación administrado. Debe haberse detenido la flota de computación o configurado [QueueUpdateStrategy](#) para poder cambiar esta configuración y proceder a la actualización.

La flota de computación debe detenerse (utilizando [pcluster update-compute-fleet](#)) para eliminar un grupo de ubicaciones administradas. Si ejecuta una actualización del clúster para eliminar un grupo de ubicaciones administrado antes de detener la flota de computación, aparecerá un mensaje de configuración no válido y la actualización no se realizará. Si se detiene la flota de computación, se garantiza que no se esté ejecutando ninguna instancia.

Ejemplos de `pcluster update-cluster`

Esta configuración se puede cambiar, pero si el cambio quiere reducir el tamaño de la cola, la flota informática debe detenerse (mediante `pcluster update-compute-fleet`) o configurarse en [QueueUpdateStrategy](#) `TERMINATE`. Una vez que la flota de procesamiento se detenga o [QueueUpdateStrategy](#) se establezca en `TERMINATE`, puede actualizar el clúster ([pcluster update-cluster](#)) para activar los cambios.

- En este ejemplo, se muestra una actualización con algunos cambios permitidos y la actualización se inicia directamente.

```
$ pcluster update-cluster --cluster-name cluster_name --cluster-config
~/./parallelcluster/test_cluster --region us-east-1
{
  "cluster": {
    "clusterName": cluster_name,
    "cloudformationStackStatus": "UPDATE_IN_PROGRESS",
    "cloudformationStackArn": stack_arn,
    "region": "us-east-1",
    "version": "3.7.0",
    "clusterStatus": "UPDATE_IN_PROGRESS"
  },
  "changeSet": [
```

```

    {
      "parameter": "HeadNode.Networking.AdditionalSecurityGroups",
      "requestedValue": [
        "sg-0cd61884c4ad11234"
      ],
      "currentValue": [
        "sg-0cd61884c4ad16341"
      ]
    }
  ]
}

```

- En este ejemplo, se muestra una actualización en seco con algunos cambios permitidos. Dryrun es útil para informar del conjunto de cambios sin iniciar la actualización.

```

$ pcluster update-cluster --cluster-name cluster_name --cluster-config
~/.parallelcluster/test_cluster --region us-east-1 --dryrun true
{
  "message": "Request would have succeeded, but DryRun flag is set.",
  "changeSet": [
    {
      "parameter": "HeadNode.Networking.AdditionalSecurityGroups",
      "requestedValue": [
        "sg-0cd61884c4ad11234"
      ],
      "currentValue": [
        "sg-0cd61884c4ad16341"
      ]
    }
  ]
}

```

- En este ejemplo se muestra una actualización con algunos cambios que bloquean la actualización.

```

$ pcluster update-cluster --cluster-name cluster_name --cluster-config
~/.parallelcluster/test_cluster --region us-east-1
{
  "message": "Update failure",
  "updateValidationErrors": [
    {
      "parameter": "HeadNode.Ssh.KeyName",
      "requestedValue": "mykey_2",

```

```

    "message": "Update actions are not currently supported for the 'KeyName'
parameter. Restore 'KeyName' value to 'jenkinsjun'. If you need this change, please
consider creating a new cluster instead of updating the existing one.",
    "currentValue": "mykey_1"
  },
  {
    "parameter": "Scheduling.SlurmQueues[queue1].ComputeResources[queue1-
t2micro].InstanceType",
    "requestedValue": "c4.xlarge",
    "message": "All compute nodes must be stopped. Stop the compute fleet with the
pcluster update-compute-fleet command",
    "currentValue": "t2.micro"
  },
  {
    "parameter": "SharedStorage[ebs1].MountDir",
    "requestedValue": "/my/very/very/long/shared_dir",
    "message": "Update actions are not currently supported for the 'MountDir'
parameter. Restore 'MountDir' value to '/shared'. If you need this change, please
consider creating a new cluster instead of updating the existing one.",
    "currentValue": "/shared"
  }
],
"changeSet": [
  {
    "parameter": "HeadNode.Networking.AdditionalSecurityGroups",
    "requestedValue": [
      "sg-0cd61884c4ad11234"
    ],
    "currentValue": [
      "sg-0cd61884c4ad16341"
    ]
  },
  {
    "parameter": "HeadNode.Ssh.KeyName",
    "requestedValue": "mykey_2",
    "currentValue": "mykey_1"
  },
  {
    "parameter": "Scheduling.SlurmQueues[queue1].ComputeResources[queue1-
t2micro].InstanceType",
    "requestedValue": "c4.xlarge",
    "currentValue": "t2.micro"
  },
  {

```

```
    "parameter": "SharedStorage[ebs1].MountDir",
    "requestedValue": "/my/very/very/long/shared_dir",
    "currentValue": "/shared"
  }
]
```

AWS ParallelCluster Personalización de AMI

Hay situaciones en las que AWS ParallelCluster es necesario crear una AMI personalizada para. En esta sección se explica qué se debe tener en cuenta al crear una AWS ParallelCluster AMI personalizada.

Puede crear una AWS ParallelCluster AMI personalizada mediante uno de los métodos siguientes:

1. Cree un [archivo de configuración de imagen de compilación](#) y a continuación, utilice la CLI `pccluster` para crear la imagen con Generador de imágenes de EC2. Este proceso es automatizado, repetible y admite la supervisión. Para obtener más información, consulte los comandos de imagen [pccluster](#).
2. Cree una instancia a partir de una AWS ParallelCluster AMI, inicie sesión en ella y realice las modificaciones manuales. Por último, utilice Amazon EC2 para crear una AMI nueva a partir de la instancia modificada. Este proceso lleva menos tiempo. Sin embargo, no es automático ni repetible, y no admite el uso de los comandos de monitoreo de imágenes CLI `pccluster`.

Para obtener más información acerca de estos métodos, consulte [Creación de una AWS ParallelCluster AMI personalizada](#).

AWS ParallelCluster Consideraciones sobre la personalización de la AMI

Independientemente de cómo cree la imagen personalizada, le recomendamos que realice pruebas de validación preliminares e incluya disposiciones para supervisar el estado de la imagen que se está creando.

Para crear una AMI personalizada mediante `pccluster`, debe crear un [archivo de configuración de generador de imágenes](#) con una sección de [Build](#) y [Image](#) que [Generador de imágenes de EC2](#) utiliza para crear la imagen personalizada. En la sección `Build` se especifica lo que Generador de imágenes necesita para crear la imagen. Esto incluye la [ParentImage](#) (imagen base) y [Components](#). Un [componente de Generador de imágenes](#) define una secuencia de pasos

necesarios para personalizar una instancia antes de crear una imagen o para probar una instancia lanzada por la imagen creada. Para ver ejemplos de AWS ParallelCluster componentes, consulte [AMI personalizadas](#). En la sección Image se especifican las propiedades de la imagen.

Cuando se le llama desde pcluster [build-image](#) para crear una imagen personalizada, Image Builder utiliza la configuración de la imagen de compilación con el AWS ParallelCluster libro de cocina para iniciar su. AWS ParallelCluster [ParentImage](#) Generador de imágenes descarga componentes, ejecuta las fases de compilación y validación, crea la AMI, lanza una instancia desde la AMI y ejecuta las pruebas. Cuando se complete el proceso, Generador de imágenes generará una nueva imagen o un mensaje de parada.

Realice pruebas de validación de componentes personalizadas

Antes de incluir un componente de Generador de imágenes en una configuración, pruébelo y válidelo mediante uno de los métodos siguientes. Como el proceso de Generador de imágenes puede tardar hasta 1 hora, le recomendamos que pruebe los componentes con antelación. Esto puede ahorrarle una cantidad considerable de tiempo.

Caso de script

Pruebe el script en una instancia en ejecución, fuera del proceso de creación de la imagen, y compruebe que el script sale con el código de salida 0.

Nombre de recurso de Amazon (ARN)

Pruebe el documento del componente en una instancia en ejecución, fuera del proceso de creación de la imagen. Para obtener una lista de requisitos, consulte [Administrador de componentes](#) en la Guía del usuario de Generador de imágenes.

Tras la validación correcta, añada el componente a la configuración de la imagen de compilación

Tras comprobar que el componente personalizado funciona, agréguelo al [archivo de configuración de la imagen de compilación](#).

Supervise el proceso de Generador de imágenes con comandos **pcluster** para facilitar la depuración

[describe-image](#)

Utilice este comando para supervisar el estado de la imagen de compilación.

[list-image-log-streams](#)

Utilice este comando para obtener los ID de los flujos de registro con los que puede recuperar los eventos de registro [get-image-log-events](#).

[get-image-log-events](#)

Utilice este comando para obtener el flujo de registro de los eventos del proceso de creación de imágenes.

Por ejemplo, puede crear eventos de imagen mediante el siguiente comando:

```
$ watch -n 1 'pcluster get-image-log-events -i <image-id> \
--log-stream-name/1 <pcluster-version> \
--query "events[*].message" | tail -n 50'
```

[get-image-stack-events](#)

Utilice este comando para recuperar los eventos de la pila que crea Generador de imágenes.

[export-image-logs](#)

Utilice este comando para guardar registros de imágenes.

Para obtener más información sobre AWS ParallelCluster los registros y Amazon CloudWatch, consulte [Registros de imágenes de compilación de los Registros de Amazon CloudWatch y Panel de Amazon CloudWatch](#).

Otras consideraciones

Nuevas AWS ParallelCluster versiones y AMI personalizadas

Si crea una AMI personalizada, debe repetir los pasos que utilizó para crear la AMI personalizada con cada nueva versión de AWS ParallelCluster .

Acciones de arranque personalizadas

Revise la [Acciones de arranque personalizadas](#) sección para determinar si las modificaciones que desee realizar se pueden programar y admitir en futuras AWS ParallelCluster versiones.

Uso de AMI personalizadas

Puede especificar las AMI personalizadas en la configuración del clúster en las secciones [Image / CustomAmi](#) y [Scheduling / SlurmQueues / - Name / Image / CustomAmi](#).

Para solucionar problemas de las advertencias de validación de AMI personalizadas, consulte [Solución de problemas con las AMI de](#) .

Inicio de instancias con ODCR (reservas de capacidad bajo demanda)

Con las [reservas de capacidad bajo demanda](#) (ODCR) puede reservar capacidad para las instancias de Amazon EC2 de su clúster en una zona de disponibilidad específica. De este modo, puede crear y administrar reservas de capacidad de forma independiente de las cuentas de facturación que ofrecen los [Savings Plans](#) o las [Instancias reservadas regionales](#).

Puede configurar reservas de capacidad bajo demanda (ODCR) open o targeted. Las ODCR abiertas cubren cualquier instancia que coincida con los atributos de la ODCR. Estos atributos son el tipo de instancia, la plataforma y la zona de disponibilidad. Debe definir de forma explícita las ODCR específicas en la configuración del clúster. Para determinar si una ODCR es open o targeted, ejecute el comando EC2 de AWS CLI [describe-capacity-reservation](#).

También puede crear una ODCR en un grupo con ubicación en clúster que se denomina [reserva de capacidad bajo demanda de un grupo con ubicación en clúster \(CPG ODCR\)](#).

Se pueden agrupar varias ODCR en un grupo de recursos. Esto se puede definir en el archivo de configuración del clúster. Para obtener más información acerca de los grupos de recursos, consulte [¿Qué son los grupos de recursos?](#) en la Guía del usuario de grupos de recursos y etiquetas.

Uso de ODCR con AWS ParallelCluster

AWS ParallelCluster admite ODCR abiertas. Al utilizar una ODCR abierta, no es necesario especificar nada en AWS ParallelCluster. Las instancias se seleccionan automáticamente para el clúster. Puede especificar un grupo de ubicación existente o elegir que AWS ParallelCluster le cree uno nuevo.

ODCR en la configuración del clúster

A partir de la versión 3.3.0 de AWS ParallelCluster, puede definir las ODCR en el archivo de configuración del clúster, sin necesidad de especificar las anulaciones de las instancias de ejecución de EC2.

Comience por crear [reservas de capacidad](#) y [grupos de recursos](#) mediante los métodos descritos en la documentación vinculada para cada uno. Debe utilizar los métodos de AWS CLI para crear

grupos de reservas de capacidad. Si usa la AWS Management Console, solo puede crear grupos de recursos basados en etiquetas o en pilas. Los grupos de recursos basados en etiquetas y en pilas no son compatibles con AWS ParallelCluster o la AWS CLI al iniciar instancias con reservas de capacidad.

Una vez que se hayan creado las reservas de capacidad y los grupos de recursos, especifíquelos en [SlurmQueues/CapacityReservationTarget](#) o [SlurmQueues/ComputeResources/CapacityReservationTarget](#), tal y como se muestra en el siguiente ejemplo de configuración de clúster. Sustituya *los valores* resaltados en rojo por valores válidos.

```
Image:
  Os: os
HeadNode:
  InstanceType: head_node_instance
  Networking:
    SubnetId: public_subnet_id
  Ssh:
    KeyName: key_name
Scheduling:
  Scheduler: scheduler
SlurmQueues:
  - Name: queue1
    Networking:
      SubnetIds:
        - private_subnet_id
ComputeResources:
  - Name: cr1
    Instances:
      - InstanceType: instance
    MaxCount: max_queue_size
    MinCount: max_queue_size
    Efa:
      Enabled: true
    CapacityReservationTarget:
      CapacityReservationResourceGroupArn: capacity_reservation_arn
```


OBSOLETO O NO RECOMENDADO: ODCR de destino con anulaciones de instancias de EC2

Warning

- A partir de la versión 3.3.0 de AWS ParallelCluster, no recomendamos este método. Esta sección sigue siendo una referencia para las implementaciones que utilizan versiones anteriores.
- Este método no es compatible con la asignación de varios tipos de instancias con Slurm.

La compatibilidad de las ODCR de `targeted` se añade en la versión 3.1.1 de AWS ParallelCluster. En esta versión, se introdujo un mecanismo que anula los parámetros `RunInstances` de EC2 y transmite información sobre la reserva para utilizarla en cada recurso de computación configurado en AWS ParallelCluster. Este mecanismo es compatible con las ODCR `targeted`. Sin embargo, cuando utilice las ODCR `targeted`, debe especificar la configuración de anulación de `run-instances`. Las ODCR específicas deben definirse explícitamente en el comando EC2 de AWS CLI [run-instances](#). Para determinar si una ODCR es `open` o `targeted`, ejecute el comando EC2 de AWS CLI [describe-capacity-reservation](#).

Se pueden agrupar varias ODCR en un grupo de recursos. Esto se puede utilizar en la anulación de instancias de ejecución para dirigirse a varias ODCR al mismo tiempo.

Si utiliza una ODCR `targeted`, puede especificar un grupo de ubicaciones. Sin embargo, también debe especificar una configuración de anulación de `run-instances`.

Suponga que AWS ha creado una ODCR `targeted` para usted o que tiene un conjunto específico de instancias reservadas. Entonces, no puede especificar un grupo de ubicación. Las reglas configuradas por AWS pueden entrar en conflicto con la configuración del grupo de ubicación. Por lo tanto, si necesita un grupo de ubicación para su solicitud, utilice una [CPG ODCR](#). En cualquier caso, también debe especificar la configuración de anulación de `run-instances`.

Si utiliza una CPG ODCR, debe especificar la configuración de anulación de `run-instances` y el mismo grupo de ubicación en la configuración del clúster.

Uso de instancias reservadas con AWS ParallelCluster


Las instancias reservadas [son diferentes](#) de las reservas de capacidad (ODCR). Hay [dos tipos](#) de instancias reservadas. Una instancia reservada regional no reserva capacidad. Una instancia reservada de zona reserva capacidad en la zona de disponibilidad especificada.

Si tiene instancias reservadas regionales, no hay reserva de capacidad y es posible que se produzcan errores de capacidad insuficiente. Si tiene instancias reservadas de zona, tiene reserva de capacidad, pero no hay parámetros de API de `run-instances` que pueda usar para especificarlas.

Las instancias reservadas son compatibles con cualquier versión de AWS ParallelCluster. No es necesario especificar nada en AWS ParallelCluster y las instancias se seleccionan automáticamente.

Al utilizar instancias reservadas de zona, puede evitar posibles errores de capacidad insuficiente omitiendo la especificación del grupo de ubicación en la configuración del clúster.

OBSOLETO O NO RECOMENDADO: utilizar la personalización de **RunInstances** en AWS ParallelCluster 3 para las reservas de capacidad bajo demanda (ODCR) **targeted**.

 Warning

- A partir de la versión 3.3.0 de AWS ParallelCluster, no recomendamos este método. Esta sección sigue siendo una referencia para las implementaciones que utilizan versiones anteriores.
- Este método no es compatible con la asignación de varios tipos de instancias con Slurm.

Puede anular los parámetros `RunInstances` de EC2 para cada recurso de computación que esté configurado en la cola de un clúster. Para ello, cree el archivo `/opt/slurm/etc/pcluster/run_instances_overrides.json` en el nodo principal del clúster con el siguiente contenido del fragmento de código:

- `${queue_name}` es el nombre de la cola a la que desea aplicar anulaciones.
- `${compute_resource_name}` es el recurso de computación al que desea aplicar anulaciones.
- `${overrides}` es un objeto JSON arbitrario que contiene una lista de anulaciones de `RunInstances` para utilizarlas en una combinación específica de cola y tipo de instancia. La sintaxis de las anulaciones debe seguir las mismas especificaciones que se documentan en una llamada boto3 a [run_instances](#).

```
{
  "${queue_name}": {
    "${compute_resource_name}": {
      ${overrides}
    }
  }
}
```

```

    },
    ...
  },
  ...
}

```

Por ejemplo, el siguiente JSON configura el grupo de ODCR `group_arn` para que se utilice en las instancias de `p4d.24xlarge` configuradas en `my-queue` y `my-compute-resource`.

```

{
  "my-queue": {
    "my-compute-resource": {
      "CapacityReservationSpecification": {
        "CapacityReservationTarget": {
          "CapacityReservationResourceGroupArn": "group_arn"
        }
      }
    }
  }
}

```

Una vez generado este archivo JSON, los daemons AWS ParallelCluster responsables del escalado del clúster utilizan automáticamente la configuración de anulación para el lanzamiento de instancias. Para confirmar que los parámetros especificados se utilizan para el aprovisionamiento de instancias, consulte los siguientes archivos de registro:

- `/var/log/parallelcluster/clustermgtd` (para capacidad estática)
- `/var/log/parallelcluster/slurm_resume.log` (para capacidad dinámica)

Si los parámetros son correctos, encontrará una entrada de registro que contiene lo siguiente:

```
Found RunInstances parameters override. Launching instances with: <parameters_list>
```

OBSOLETO O NO RECOMENDADO: cree un clúster con reservas de capacidad bajo demanda (ODCR) **targeted**

Warning

- A partir de la versión 3.3.0 de AWS ParallelCluster, no recomendamos este método. Esta sección sigue siendo una referencia para las implementaciones que utilizan versiones anteriores.
- Este método no es compatible con [Asignación de varios tipos de instancias con Slurm](#).

1. Cree un grupo de recursos para agrupar la capacidad.

```
$ aws resource-groups create-group --name EC2CRGroup \
  --configuration '{"Type":"AWS::EC2::CapacityReservationPool"}'
  '{"Type":"AWS::ResourceGroups::Generic", "Parameters": [{"Name": "allowed-
resource-types", "Values": ["AWS::EC2::CapacityReservation"]}]}'
```

Note

Un grupo de recursos no admite recursos que otras cuentas compartan.

Si la ODCR específica la comparte otra cuenta, no es necesario crear un grupo de recursos. Utilice CapacityReservationId en lugar de un grupo de recursos en el paso 3.

```
#!/bin/bash
set -e

# Override run_instance attributes
cat > /opt/slurm/etc/pcluster/run_instances_overrides.json << EOF
{
  "my-queue": {
    "my-compute-resource": {
      "CapacityReservationSpecification": {
        "CapacityReservationTarget": {
          "CapacityReservationId": "cr-abcdef01234567890"
        }
      }
    }
  }
}
```

```

    }
  }
EOF

```

Agregue reservas de capacidad al grupo de recursos. Cada vez que cree una ODCR nueva, agréguela a la reserva de grupo. Sustituya *ACCOUNT_ID* por su ID de cuenta, *PLACEHOLDER_CAPACITY_RESERVATION* por su ID de reserva de capacidad y *REGION_ID* por su ID de Región de AWS (por ejemplo, us-east-1).

```

$ aws resource-groups group-resources --region REGION_ID --group EC2CRGroup \
  --resource-arns arn:aws:ec2:REGION_ID:ACCOUNT_ID:capacity-
reservation/PLACEHOLDER_CAPACITY_RESERVATION

```

Cree un documento de política en su equipo local. Sustituya *ACCOUNT_ID* por su ID de cuenta y *REGION_ID* por su ID de Región de AWS (por ejemplo, us-east-1).

```

cat > policy.json << EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "RunInstancesInCapacityReservation",
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:REGION_ID:ACCOUNT_ID:capacity-reservation/*",
        "arn:aws:resource-groups:REGION_ID:ACCOUNT_ID:group/*"
      ]
    }
  ]
}
EOF

```

2. Cree la política de IAM en su Cuenta de AWS utilizando el archivo json que ha creado.

```

$ aws iam create-policy --policy-name RunInstancesCapacityReservation --policy-
document file://policy.json

```

3. Cree el siguiente script posterior a la instalación de forma local en la instancia y llámelo **postinstall.sh**.

Sustituya **ACCOUNT_ID** por su ID de Cuenta de AWS y **REGION_ID** por su ID de Región de AWS (por ejemplo, us-east-1).

```
#!/bin/bash
set -e

# Override run_instance attributes
cat > /opt/slurm/etc/pcluster/run_instances_overrides.json << EOF
{
  "my-queue": {
    "my-compute-resource": {
      "CapacityReservationSpecification": {
        "CapacityReservationTarget": {
          "CapacityReservationResourceGroupArn": "arn:aws:resource-
groups:REGION_ID:ACCOUNT_ID:group/EC2CRGroup"
        }
      }
    }
  }
}
EOF
```

Cargue el archivo en un bucket de Amazon S3. Sustituya **S3_NAME_BUCKET** por el nombre del bucket específico de Amazon S3.

```
$ aws s3 mb s3://S3_NAME_BUCKET
aws s3 cp postinstall.sh s3://S3_NAME_BUCKET/postinstall.sh
```

4. Cree la configuración del clúster local y sustituya los marcadores de posición por sus propios valores.

```
Region: REGION_ID
Image:
  Os: alinux2
HeadNode:
  InstanceType: c5.2xlarge
  Ssh:
    KeyName: YOUR_SSH_KEY
  Iam:
```

```

S3Access:
  - BucketName: S3_NAME_BUCKET
AdditionalIamPolicies:
  - Policy: arn:aws:iam::ACCOUNT_ID:policy/RunInstancesCapacityReservation
## This post-install script is executed after the node is configured.
## It is used to install scripts at boot time and specific configurations
## In the script below we are overriding the calls to RunInstance to force
## the provisioning of our my-queue partition to go through
## the On-Demand Capacity Reservation
CustomActions:
  OnNodeConfigured:
    Script: s3://S3_NAME_BUCKET/postinstall.sh
Networking:
  SubnetId: YOUR_PUBLIC_SUBNET_IN_TARGET_AZ

Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: my-queue
      ComputeResources:
        - MinCount: 0
          MaxCount: 100
          InstanceType: p4d.24xlarge
          Name: my-compute-resource
          Efa:
            Enabled: true
      Networking:
        ## PlacementGroup:
        ##   Enabled: true ## Keep PG disabled if using targeted ODCR
        SubnetIds:
          - YOUR_PRIVATE_SUBNET_IN_TARGET_AZ

```

5. Cree el clúster.

Utilice el siguiente comando para crear el clúster. Sustituya *cluster-config.yaml* por el nombre del archivo de configuración, *cluster-d1* por el nombre del clúster y *REGION_ID* por el ID de la región (por ejemplo, us-east-1).

```

$ pcluster create-cluster --cluster-configuration cluster-config.yaml --cluster-
name cluster-d1 --region REGION_ID

```

Una vez creado el clúster, el script posterior a la instalación se ejecuta en el nodo principal. El script crea el archivo `run_instances_overrides.json` y anula las llamadas a `RunInstances` para forzar el aprovisionamiento de la partición a través de la reserva de capacidad bajo demanda.

Los daemons AWS ParallelCluster responsables del escalado de los clústeres utilizan automáticamente esta configuración para las nuevas instancias que se lanzan. Para confirmar que los parámetros especificados se utilizan para el aprovisionamiento de instancias, puede consultar los siguientes archivos de registro:

- `/var/log/parallelcluster/clustermgtd` (para capacidad estática: [MinCount](#) > 0)
- `/var/log/parallelcluster/slurm_resume.log` (para capacidad dinámica)

Si los parámetros son correctos, encontrará una entrada de registro que contiene lo siguiente.

```
Found RunInstances parameters override. Launching instances with: <parameters_list>
```

Actualización de anulaciones de **RunInstances**

Puede actualizar la configuración JSON generada en cualquier momento sin detener la flota de computación. Una vez aplicados los cambios, todas las instancias nuevas se lanzan con la configuración actualizada. Si necesita aplicar la configuración actualizada a los nodos en ejecución, recicle los nodos forzando la finalización de una instancia y espere a que AWS ParallelCluster sustituya esos nodos. Para ello, puede finalizar la instancia desde la consola EC2 o AWS CLI, o bien configurar los nodos de Slurm en un estado DOWN o DRAIN.

Utilice el siguiente comando para establecer el nodo de Slurm en DOWN o DRAIN.

```
$ scontrol update nodename=my-queue-dy-my-compute-resource-1 state=down  
reason=your_reason  
scontrol update nodename=my-queue-dy-my-compute-resource-1 state=drain  
reason=your_reason
```

Parqueo de AMI y sustitución de instancias de EC2

Para garantizar que todos los nodos de computación de clúster lanzados dinámicamente se comporten de forma coherente, AWS ParallelCluster desactiva las actualizaciones automáticas del

sistema operativo de las instancias de clúster. Además, se crea un conjunto específico de AMI de AWS ParallelCluster para cada versión de AWS ParallelCluster y su CLI asociada. Este conjunto específico de AMI permanece inalterado y solo es compatible con la versión AWS ParallelCluster para la que se crearon. AWS ParallelCluster Las AMI de las versiones publicadas no se actualizan.

Sin embargo, debido a problemas de seguridad emergentes, es posible que los clientes deseen añadir parches a estas AMI y, a continuación, actualizar sus clústeres con la AMI parcheada. Esto se ajusta al [modelo de responsabilidad compartida de AWS ParallelCluster](#).

Para ver el conjunto específico de AMI de AWS ParallelCluster compatibles con la versión de CLI de AWS ParallelCluster que está utilizando actualmente, ejecute:

```
$ pcluster version
$ pcluster list-official-images
```

El nodo principal de AWS ParallelCluster es una instancia estática y puede actualizarla manualmente. El reinicio del nodo principal es totalmente compatible a partir de la versión 3.0.0 de AWS ParallelCluster.

Si sus instancias tienen almacenes de instancias efímeros, no olvide guardar los datos del almacén de instancias antes de llevar a cabo las actualizaciones manuales. Para obtener más información, consulte la configuración del clúster [HeadNode/LocalStorage/EphemeralVolume](#) y los [tipos de instancias con volúmenes del almacén de instancias](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

Los nodos de computación son instancias efímeras. De forma predeterminada, solo se puede acceder a ellos desde el nodo principal. A partir de la versión 3.0.0 de AWS ParallelCluster, puede actualizar la AMI asociada a las instancias de computación modificando el parámetro [Scheduling/SlurmQueues/Image/CustomAmi](#) y ejecutando el comando [pcluster update-cluster](#), después de detener la flota de computación con [pcluster update-compute-fleet](#):

```
$ pcluster update-compute-fleet-status --status STOP_REQUESTED
```

Es posible automatizar la creación de una AMI personalizada actualizada para los nodos de computación mediante uno de los siguientes métodos:

- Use el comando [pcluster build-image](#) con una [Build/ParentImage](#) actualizada.
- Ejecute la compilación con [Build/UpdateOsPackages/Enabled:true](#).

Actualización o sustitución de la instancia del nodo principal

En algunas circunstancias, es posible que tenga que reiniciar el nodo principal. Por ejemplo, es necesario cuando actualiza manualmente el sistema operativo o cuando hay una [retirada programada de una instancia de AWS](#) que impone el reinicio de la instancia del nodo principal.

Si la instancia no tiene unidades efímeras, puede detenerla e iniciarla de nuevo en cualquier momento. En el caso de una retirada programada, al iniciar la instancia detenida, se realiza una migración para usar el nuevo hardware.

Del mismo modo, puede detener e iniciar manualmente una instancia que no tenga almacenes de instancias. En este caso y en otros casos de instancias sin volúmenes efímeros, continúe hasta [Detención e inicio del nodo principal de un clúster](#).

Si la instancia tiene unidades efímeras y se ha detenido, se perderán los datos del almacén de instancias. Puede determinar si el tipo de instancia utilizado para el nodo principal tiene almacenes de instancias a partir de la tabla que se encuentra en los [volúmenes de almacenes de instancias](#).

Guarde los datos de unidades efímeras

A partir de la versión 3.0.0 de AWS ParallelCluster, el reinicio del nodo principal es totalmente compatible con todos los tipos de instancias. Sin embargo, si las instancias tienen una unidad efímera, se perderán sus datos. Siga los siguientes pasos para conservar los datos antes de que se reinicie el nodo principal.

Para comprobar si tiene datos que deba conservar, consulte el contenido de la carpeta [EphemeralVolume/MountDir](#) (/scratch de forma predeterminada).

Puede transferir los datos al volumen raíz o a los sistemas de almacenamiento compartido asociados al clúster, como Amazon FSx, Amazon EFS o Amazon EBS. Tenga en cuenta que la transferencia de datos al almacenamiento remoto puede conllevar costes adicionales.

Después de guardar los datos, continúe hasta [Detención e inicio del nodo principal de un clúster](#).

Detención e inicio del nodo principal de un clúster

1. Compruebe que no haya ningún trabajo en ejecución en el clúster.

Cuando utilice un programador Slurm:

- Si no se especifica la opción `--no-requeue` de `sbatch`, los trabajos en ejecución se vuelven a poner en cola.
- Si se especifica la opción `--no-requeue`, se produce un error en los trabajos en ejecución.

2. Solicite la detención de la flota de computación de un clúster:

```
$ pcluster update-compute-fleet --cluster-name cluster-name --status STOP_REQUESTED
{
  "status": "STOP_REQUESTED",
  ...
}
```

3. Espere hasta que el estado de la flota de computación sea STOPPED:

```
$ pcluster update-compute-fleet --cluster-name cluster-name --status STOP_REQUESTED
{
  "status": "STOPPED",
  ...
}
```

4. Para las actualizaciones manuales con un reinicio del sistema operativo o un reinicio de la instancia, puede usar la AWS Management Console o la AWS CLI. A continuación, se muestra un ejemplo de cómo se utiliza AWS CLI.

```
# Retrieve head node instance id
$ pcluster describe-cluster --cluster-name cluster-name --status STOP_REQUESTED
{
  "headNode": {
    "instanceId": "i-1234567890abcdef0",
    ...
  },
  ...
}
# stop and start the instance
$ aws ec2 stop-instances --instance-ids 1234567890abcdef0
{
  "StoppingInstances": [
    {
      "CurrentState": {
        "Name": "stopping"
        ...
      },
    },
  ],
}
```

```

    "InstanceId": "i-1234567890abcdef0",
    "PreviousState": {
      "Name": "running"
      ...
    }
  ]
}
$ aws ec2 start-instances --instance-ids 1234567890abcdef0
{
  "StartingInstances": [
    {
      "CurrentState": {
        "Name": "pending"
        ...
      },
      "InstanceId": "i-1234567890abcdef0",
      "PreviousState": {
        "Name": "stopped"
        ...
      }
    }
  ]
}

```

5. Inicie la flota de computación del clúster:

```

$ pcluster update-compute-fleet --cluster-name cluster-name --status
START_REQUESTED
{
  "status": "START_REQUESTED",
  ...
}

```

Sistemas operativos

AWS ParallelCluster es compatible con Amazon Linux 2, CentOS 7, Ubuntu 22.04, Ubuntu 2004, Red Hat Enterprise Linux 8 (RHEL8), Rocky 8, Red Hat Enterprise Linux 9 (RHEL9), Rocky 9. AWS ParallelCluster ofrece AMI prediseñadas para determinados sistemas operativos. Para obtener más información sobre las AMI proporcionadas, consulte. AWS ParallelCluster [Sección de Image](#)

Consideraciones sobre el sistema operativo

Ubuntu 2.04

Ubuntu 2204 requiere claves más seguras para SSH y no admite claves de RSA de forma predeterminada. Genere una clave ed25519 y úsela para crear el clúster.

Ubuntu 2204 no se puede actualizar a la última versión del kernel porque no hay un cliente Fsx para ese kernel.

RHEL 8

RedHat Enterprise Linux 8.7 (rhel8) se agrega a partir de la versión 3.6.0. AWS ParallelCluster Si configura el clúster para usar rhel8, el costo bajo demanda de cualquier tipo de instancia es mayor que si configura el clúster para usar otros sistemas operativos compatibles.

Para obtener más información sobre los precios, consulte [Precios bajo demanda](#) y [¿Cómo se ofrece y se cotiza Red Hat Enterprise Linux en Amazon EC2?](#).

Rocky 8

AWS ParallelCluster La versión 3.8.0 es compatible con Rocky Linux 8, pero las AMI Rocky Linux 8 prediseñadas (para arquitecturas x86 y ARM) no están disponibles. AWS ParallelCluster La versión 3.8.0 permite crear clústeres con Rocky Linux 8 mediante AMI personalizadas que utilicen la propiedad. [CustomAmi](#) Para obtener más información sobre cómo crear AMI personalizadas, consulte. [AWS ParallelCluster Personalización de AMI](#)

Para crear su AMI personalizada a partir de una AMI básica de Rocky Linux 8, puede considerar suscribirse a las AMI de [Rocky Linux 8](#) disponibles en AWS [Marketplace](#). Asegúrese de revisar los precios y los costos de suscripción de las AMI Rocky Linux 8 en AWS Marketplace. Como alternativa, también puede utilizar las AMI [oficiales de Rocky Linux 8 como AMI](#) base.

Centos7

[Gdrcopy](#) ha eliminado a Centos7 de su matriz de soporte de sistemas operativos. Esto significa que gdrcopy 2.3.1 es la última versión compatible con este sistema operativo. Debe fijar las versiones NVIDIA y gdrcopy para Centos7, ya que las versiones más recientes del controlador de código abierto de NVIDIA (OpenRM, es decir, 535.129.03+) no son compatibles con esta versión de gdrcopy. A partir de la ParallelCluster versión 3.8.0, nuestras AMI oficiales de Centos7 se lanzarán con gdrcopy 2.3.1 y el controlador NVIDIA 535.129.03.

Rocky 9

AWS ParallelCluster La versión 3.9.0 es compatible con Rocky Linux 9, pero las AMI de Rocky Linux 9 prediseñadas (para arquitecturas x86 y ARM) no están disponibles. AWS ParallelCluster La versión 3.9.0 permite crear clústeres con Rocky Linux 9 mediante AMI personalizadas que utilicen la propiedad. [CustomAmi](#) Para obtener más información sobre la creación de AMI personalizadas, consulte [Personalización de AWS ParallelCluster AMI](#). Para crear su AMI personalizada a partir de una AMI básica de Rocky Linux 9, también puede utilizar las AMI [oficiales de Rocky Linux 9 como AMI](#) base. La compilación de la AMI personalizada de Rocky Linux 9 puede fallar si la AMI base no tiene el núcleo más reciente. Para actualizar el núcleo antes de compilar la AMI:

- [Lanza una instancia con un ID de AMI de rocky9 desde aquí: https://rockylinux.org/cloud-images/](https://rockylinux.org/cloud-images/)
- entra en la instancia y ejecuta el siguiente comando: `sudo yum -y update`
- Crea una imagen a partir de la instancia para usarla como ParentImage

Referencia para AWS ParallelCluster

Temas

- [Comandos de la CLI de AWS ParallelCluster](#)
- [Archivos de configuración](#)
- [Referencia de la API de AWS ParallelCluster](#)
- [API de la biblioteca Python de AWS ParallelCluster](#)

Comandos de la CLI de AWS ParallelCluster

`pcluster` es el comando principal de la CLI de AWS ParallelCluster. `pcluster` se utiliza para lanzar y administrar clústeres de HPC en Nube de AWS y para crear y administrar imágenes AMI personalizadas.

`pcluster3-config-convert` se utiliza para convertir las configuraciones de clústeres del formato de AWS ParallelCluster versión 2 al formato de AWS ParallelCluster versión 3.

```
pcluster [-h] ( build-image | configure |
               create-cluster | dcv-connect |
               delete-cluster | delete-cluster-instances | delete-image |
               describe-cluster | describe-cluster-instances |
               describe-compute-fleet | describe-image |
               export-cluster-logs | export-image-logs |
               get-cluster-log-events | get-cluster-stack-events |
               get-image-log-events | get-image-stack-events |
               list-cluster-log-streams | list-clusters |
               list-images | list-image-log-streams | list-official-images |
               ssh | update-cluster |
               update-compute-fleet | version ) ...
pcluster3-config-convert [-h] [-t CLUSTER_TEMPLATE]
                        [-c CONFIG_FILE]
                        [--force-convert]
                        [-o OUTPUT_FILE]
```

Temas

- [pcluster](#)
- [pcluster3-config-convert](#)

pcluster

`pcluster` es el comando AWS ParallelCluster CLI principal. Puede usar `pcluster` para lanzar y administrar clústeres de HPC en la Nube de AWS.

`pcluster` escribe los registros de sus comandos en archivos `pcluster.log.#` en `/home/user/.parallelcluster/`. Para obtener más información, consulte [Registros de CLI de pcluster](#).

Para poder utilizar `pcluster`, debe tener un rol de IAM con los [permisos](#) necesarios para ejecutarlo.

```
pcluster [-h]
```

Argumentos

`pcluster` *command*

Posibles opciones: [build-image](#) [configure](#) [create-cluster](#) [dcv-connect](#) [delete-cluster](#) [delete-cluster-instances](#) [delete-image](#) [describe-cluster](#) [describe-cluster-instances](#) [describe-compute-fleet](#) [describe-image](#) [export-cluster-logs](#) [export-image-logs](#) [get-cluster-log-events](#) [get-cluster-stack-events](#) [get-image-log-events](#) [get-image-stack-events](#) [list-clusters](#) [list-cluster-log-streams](#) [list-images](#) [list-image-log-streams](#) [list-official-images](#) [ssh](#) [update-cluster](#) [update-compute-fleet](#) [version](#)

Subcomandos:

Temas

- [pcluster build-image](#)
- [pcluster configure](#)
- [pcluster create-cluster](#)
- [pcluster dcv-connect](#)
- [pcluster delete-cluster](#)
- [pcluster delete-cluster-instances](#)
- [pcluster delete-image](#)
- [pcluster describe-cluster](#)

- [pcluster describe-cluster-instances](#)
- [pcluster describe-compute-fleet](#)
- [pcluster describe-image](#)
- [pcluster export-cluster-logs](#)
- [pcluster export-image-logs](#)
- [pcluster get-cluster-log-events](#)
- [pcluster get-cluster-stack-events](#)
- [pcluster get-image-log-events](#)
- [pcluster get-image-stack-events](#)
- [pcluster list-clusters](#)
- [pcluster list-cluster-log-streams](#)
- [pcluster list-images](#)
- [pcluster list-image-log-streams](#)
- [pcluster list-official-images](#)
- [pcluster ssh](#)
- [pcluster update-cluster](#)
- [pcluster update-compute-fleet](#)
- [pcluster version](#)

pcluster build-image

Cree una AWS ParallelCluster imagen personalizada en la región especificada.

```
pcluster build-image [-h]
    --image-configuration IMAGE_CONFIGURATION
    --image-id IMAGE_ID
    [--debug]
    [--dryrun DRYRUN]
    [--query QUERY]
    [--region REGION]
    [--rollback-on-failure ROLLBACK_ON_FAILURE]
    [--suppress-validators SUPPRESS_VALIDATORS [SUPPRESS_VALIDATORS ...]]
    [--validation-failure-level {INFO,WARNING,ERROR}]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster build-image`.

--image-configuration, -c *IMAGE_CONFIGURATION*

Especifica el archivo de configuración de la imagen como un documento YAML.

--image-id, -i *IMAGE_ID*

Especifica el identificador de la imagen que se va a crear.

--debug

Activar el registro de depuración.

--dryrun *DRYRUN*

Cuando se establece en `true`, el comando realiza la validación sin crear ningún recurso. Puede usarlo para validar la configuración de la imagen. (el valor predeterminado es `false`).

--query *QUERY*

Consulta JMESPath que se realizará en la salida.

--region, -r *REGION*

Especifica el Región de AWS que se va a utilizar. Región de AWS Debe especificarse mediante el ajuste [Región](#) del archivo de configuración de la imagen, la variable de `AWS_DEFAULT_REGION` entorno, el `region` ajuste de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

--rollback-on-failure *ROLLBACK_ON_FAILURE*

Si se establece en `true`, inicia automáticamente una reversión de la pila de la imagen en caso de que se produzca un error (el valor predeterminado es `false`).

--suppress-validators *SUPPRESS_VALIDATORS* [*SUPPRESS_VALIDATORS ...*]

Identifica uno o más validadores de configuración para suprimirlos.

Formato: `(ALL|type:[A-Za-z0-9]+)`

--validation-failure-level {INFO,WARNING,ERROR}

Especifica el nivel mínimo de validación que provocará un error en la creación (el valor predeterminado es `ERROR`).

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.2:

```
$ pcluster build-image --image-configuration image-config.yaml --image-id custom-  
alinux2-image  
{  
  "image": {  
    "imageId": "custom-alinux2-image",  
    "imageBuildStatus": "BUILD_IN_PROGRESS",  
    "cloudformationStackStatus": "CREATE_IN_PROGRESS",  
    "cloudformationStackArn": "arn:aws:cloudformation:us-east-1:123456789012:stack/  
custom-alinux2-image/1234abcd-56ef-78gh-90ij-abcd1234efgh",  
    "region": "us-east-1",  
    "version": "3.1.2"  
  }  
}
```

Warning

`pcluster build-image` utiliza la VPC predeterminada. Si se ha eliminado la VPC predeterminada, quizás mediante AWS Control Tower una zona de AWS aterrizaje, se debe especificar el ID de subred en el archivo de configuración de la imagen. Para obtener más información, consulte [SubnetId](#)

pcluster configure

Inicia un asistente de configuración interactivo para AWS ParallelCluster la versión 3. Para obtener más información, consulte [Configure y cree un clúster con la interfaz de línea de AWS ParallelCluster comandos](#).

```
pcluster configure [-h]  
                --config CONFIG  
                [--debug]  
                [--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster configure`.

--config *CONFIG*

Ruta de la salida del archivo de configuración generado.

--debug

Activar el registro de depuración.

--region, -r *REGION*

Especifica el Región de AWS que se va a utilizar. La región debe especificarse mediante el ajuste [Región](#) del archivo de configuración de la imagen, la variable de entorno `AWS_DEFAULT_REGION`, el ajuste `region` de la sección `[default]` del archivo `~/.aws/config` o el parámetro `--region`.

pcluster create-cluster

Creación de un AWS ParallelCluster clúster.

```
pcluster create-cluster [-h]
                        --cluster-configuration CLUSTER_CONFIGURATION
                        --cluster-name CLUSTER_NAME
                        [--debug]
                        [--dryrun DRYRUN]
                        [--query QUERY]
                        [--region REGION]
                        [--rollback-on-failure ROLLBACK_ON_FAILURE]
                        [--suppress-validators SUPPRESS_VALIDATORS [SUPPRESS_VALIDATORS ...]]
                        [--validation-failure-level {INFO,WARNING,ERROR}]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster create-cluster`.

--cluster-configuration, -c *CLUSTER_CONFIGURATION*

Especifica el archivo YAML de configuración del clúster.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster que se va a crear.

El nombre debe empezar por un carácter alfabético. El nombre puede tener un máximo de 60 caracteres. Si la contabilidad de Slurm está habilitada, el nombre puede tener hasta 40 caracteres.

Los caracteres válidos son: a-z, A-Z, 0-9 y - (guion).

--debug

Habilita del registro de depuración.

--dryrun *DRYRUN*

Cuando se establece en `true`, el comando realiza la validación sin crear ningún recurso. Puede usarlo para validar la configuración del clúster. (el valor predeterminado es `false`).

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el Región de AWS que se va a utilizar. Región de AWS Debe especificarse mediante la [Region](#) configuración del archivo de configuración del clúster, la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

--rollback-on-failure *ROLLBACK_ON_FAILURE*

Si se establece en `true`, inicia automáticamente una reversión de la pila del clúster en caso de que se produzcan errores (el valor predeterminado es `true`).

--suppress-validators *SUPPRESS_VALIDATORS* [*SUPPRESS_VALIDATORS* ...]

Identifica uno o más validadores de configuración para suprimirlos.

Formato: (ALL[type:[A-Za-z0-9]+])

--validation-failure-level {INFO,WARNING,ERROR}

Especifica el nivel mínimo de validación que provocará un error en la creación (el valor predeterminado es `ERROR`).

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster create-cluster -c cluster-config.yaml -n cluster-v3
{
```

```

"cluster": {
  "clusterName": "cluster-v3",
  "cloudformationStackStatus": "CREATE_IN_PROGRESS",
  "cloudformationStackArn": "arn:aws:cloudformation:us-east-1:123456789012:stack/
cluster-v3/1234abcd-56ef-78gh-90ij-abcd1234efgh",
  "region": "us-east-1",
  "version": "3.1.4",
  "clusterStatus": "CREATE_IN_PROGRESS"
}
}

```

pcluster dcv-connect

Permite conectarse al nodo principal a través de una sesión interactiva mediante NICE DCV.

```

pcluster dcv-connect [-h]
                    --cluster-name CLUSTER_NAME
                    [--debug]
                    [--key-path KEY_PATH]
                    [--region REGION]
                    [--show-url]

```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster dcv-connect`.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--debug

Habilita del registro de depuración.

--key-path *KEY_PATH*

Especifica la ruta de la clave SSH que se utilizará para la conexión.

--region, -r *REGION*

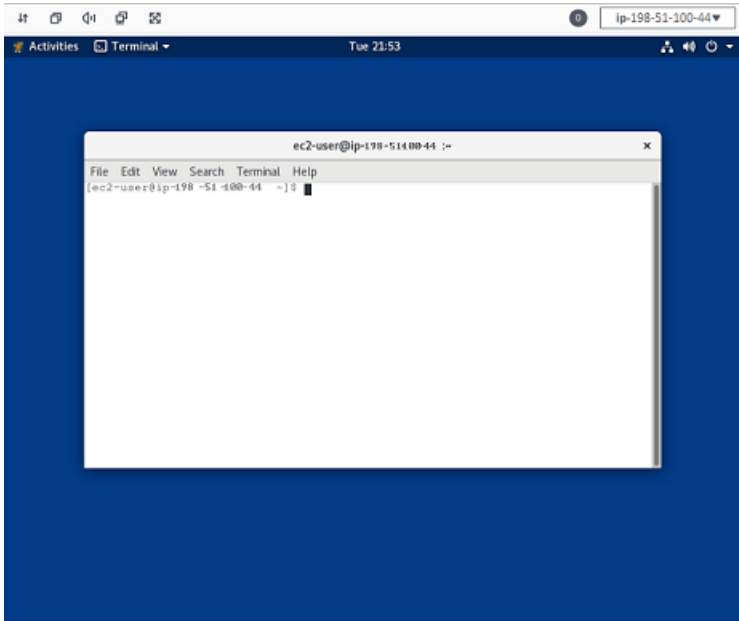
Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

--show-url

Imprime la URL que se usaría para la conexión DCV y se cierra.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster dcv-connect -n cluster-3Dcv -r us-east-1 --key-path /home/user/.ssh/key.pem
```



pcluster delete-cluster

Inicia la eliminación de un clúster.

```
pcluster delete-cluster [-h]
                        --cluster-name CLUSTER_NAME
                        [--debug]
                        [--query QUERY]
                        [--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster delete-cluster`.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--debug

Habilita del registro de depuración.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. La región debe especificarse mediante la variable de entorno `AWS_DEFAULT_REGION`, el ajuste `region` de la sección `[default]` del archivo `~/.aws/config` o el parámetro `--region`.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster delete-cluster -n cluster-v3
{
  "cluster": {
    "clusterName": "cluster-v3",
    "cloudformationStackStatus": "DELETE_IN_PROGRESS",
    "cloudformationStackArn": "arn:aws:cloudformation:us-east-1:123456789012:stack/
cluster-v3/1234abcd-56ef-78gh-90ij-abcd1234efgh",
    "region": "us-east-1",
    "version": "3.1.4",
    "clusterStatus": "DELETE_IN_PROGRESS"
  }
}
```

pcluster delete-cluster-instances

Fuerce la finalización de todos los nodos de computación del clúster. Esto no funciona con AWS Batch clústeres.

```
pcluster delete-cluster-instances [-h]
    --cluster-name CLUSTER_NAME
    [--debug]
    [--force FORCE]
    [--query QUERY]
```



```
[--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster delete-cluster-instances`.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--debug

Habilita del registro de depuración.

--force *FORCE*

Cuando se establece en `true`, fuerza la eliminación ignorando los errores de validación. (el valor predeterminado es `false`).

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el Región de AWS que se va a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

```
$ pcluster delete-cluster-instances -n cluster-v3
```

pcluster delete-image

Inicia la eliminación de la AWS ParallelCluster imagen personalizada.

```
pcluster delete-image [-h]
                      --image-id IMAGE_ID
                      [--debug]
                      [--force FORCE]
                      [--query QUERY]
                      [--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster delete-image`.

--image-id, -i *IMAGE_ID*

Especifica el identificador de la imagen que se va a eliminar.

--debug

Habilita del registro de depuración.

--force *FORCE*

Cuando se establece en `true`, fuerza la eliminación en caso de que haya instancias que usen la AMI o si la AMI es compartida (el valor predeterminado es `false`).

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el Región de AWS que se va a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster delete-image --image-id custom-alinux2-image
{
  "image": {
    "imageId": "custom-alinux2-image",
    "imageBuildStatus": "DELETE_IN_PROGRESS",
    "region": "us-east-1",
    "version": "3.1.4"
  }
}
```

pcluster describe-cluster

Obtenga información detallada sobre un clúster.

```
pcluster describe-cluster [-h]
                        --cluster-name CLUSTER_NAME
                        [--debug]
                        [--query QUERY]
                        [--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster describe-cluster`.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--debug

Habilita del registro de depuración.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplos que utilizan AWS ParallelCluster la versión 3.1.4:

Describe los detalles del clúster:

```
$ pcluster describe-cluster -n cluster-v3
{
  "creationTime": "2022-07-12T17:19:16.101Z",
  "headNode": {
    "launchTime": "2022-07-12T17:22:21.000Z",
    "instanceId": "i-1234567890abcdef0",
    "publicIpAddress": "198.51.100.44",
    "instanceType": "t2.micro",
    "state": "running",
    "privateIpAddress": "192.0.2.0.196"
```

```

},
"loginNodes": {
  "status": "active",
  "address": "8af2145440569xyz.us-east-1.amazonaws.com",
  "scheme": "internet-facing|internal",
  "healthyNodes": 3,
  "unhealthyNodes": 0
},
"version": "3.1.4",
"clusterConfiguration": {
  "url": "https://parallelcluster-e5ca74255d6c3886-v1-do-not-delete..."
},
"tags": [
  {
    "value": "3.1.4",
    "key": "parallelcluster:version"
  }
],
"cloudFormationStackStatus": "CREATE_COMPLETE",
"clusterName": "cluster-v3",
"computeFleetStatus": "RUNNING",
"cloudformationStackArn": "arn:aws:cloudformation:us-east-1:123456789012:stack/
cluster-v3/1234abcd-56ef-78gh-90ij-abcd1234efgh",
"lastUpdatedTime": "2022-07-12T17:19:16.101Z",
"region": "us-east-1",
"clusterStatus": "CREATE_COMPLETE"
}

```

Use `describe-cluster` para recuperar la configuración del clúster:

```

$ curl -o - $(pcluster describe-cluster -n cluster-v3 --query clusterConfiguration.url
| xargs echo)
Region: us-east-1
Image:
  Os: alinux2
HeadNode:
  InstanceType: t2.micro
  Networking:
    SubnetId: subnet-abcdef01234567890
  Ssh:
    KeyName: adpc
  Iam:
    S3Access:

```

```

    - BucketName: cluster-v3-bucket
      KeyName: logs
      EnableWriteAccess: true
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue1
      ComputeResources:
        - Name: t2micro
          InstanceType: t2.micro
          MinCount: 0
          MaxCount: 10
  Networking:
    SubnetIds:
      - subnet-021345abcdef6789

```

pcluster describe-cluster-instances

Describe las instancias de un clúster.

```

pcluster describe-cluster-instances [-h]
    --cluster-name CLUSTER_NAME
    [--debug]
    [--next-token NEXT_TOKEN]
    [--node-type {HeadNode,ComputeNode,LoginNode}]
    [--query QUERY]
    [--queue-name QUEUE_NAME]
    [--region REGION]

```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster describe-cluster-instances`.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--debug

Habilita del registro de depuración.

--next-token *NEXT_TOKEN*

Especifica el token que se va a utilizar para las solicitudes paginadas.

--node-type {HeadNode,ComputeNode,LoginNode}

Especifica los tipos de nodos que se van a mostrar. Los valores admitidos son HeadNode, ComputeNode y LoginNode. Si no se especifica este parámetro, se describen las instancias HeadNode, ComputeNode y LoginNode.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--queue-name *QUEUE_NAME*

Especifica el nombre de la cola que se va a mostrar. Si no se especifica este parámetro, se describen las instancias de todas las colas.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de AWS_DEFAULT_REGION entorno, la region configuración de la [default] sección del ~/.aws/config archivo o el --region parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster describe-cluster-instances -n cluster-v3
{
  "instances": [
    {
      "launchTime": "2022-07-12T17:22:21.000Z",
      "instanceId": "i-1234567890abcdef0",
      "publicIpAddress": "198.51.100.44",
      "instanceType": "t2.micro",
      "state": "running",
      "nodeType": "HeadNode",
      "privateIpAddress": "192.0.2.0.196"
    },
    {
      "launchTime": "2022-07-12T17:37:42.000Z",
      "instanceId": "i-021345abcdef6789",
      "queueName": "queue1",
      "publicIpAddress": "198.51.100.44",
      "instanceType": "t2.micro",
```

```
    "state": "pending",
    "nodeType": "ComputeNode",
    "privateIpAddress": "192.0.2.0.196"
  }
]
}
```

pcluster describe-compute-fleet

Describe el estado de la flota de computación.

```
pcluster describe-compute-fleet [-h]
    --cluster-name CLUSTER_NAME
    [--debug]
    [--query QUERY]
    [--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster describe-compute-fleet`.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--debug

Habilita del registro de depuración.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster describe-compute-fleet -n pcluster-v3
{
```

```
"status": "RUNNING",
"lastStatusUpdatedTime": "2022-07-12T17:24:26.000Z"
}
```

pcluster describe-image

Obtenga información detallada sobre una imagen.

```
pcluster describe-image [-h]
                        --image-id IMAGE_ID
                        [--debug]
                        [--query QUERY]
                        [--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster describe-image`.

--image-id, -i *IMAGE_ID*

Especifica el identificador de la imagen.

--debug

Habilita del registro de depuración.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.2:

```
$ pcluster describe-image --image-id custom-linux2-image
{
  "imageConfiguration": {
    "url": "https://parallelcluster-1234abcd5678-v1-do-not-delete.../configs/image-
config.yaml"
```



```

},
"imageId": "custom-alinux2-image",
"creationTime": "2022-04-05T20:23:07.000Z"
"imageBuildStatus": "BUILD_COMPLETE",
"region": "us-east-1",
"ec2AmiInfo": {
  "amiName": "custom-alinux2-image 2022-04-05T19-55-22.518Z",
  "amiId": "ami-1234abcd5678efgh",
  "description": "AWS ParallelCluster AMI for alinux2,
kernel-4.14.268-205.500.amzn2.x86_64, lustre-2.10.8-5.amzn2.x86_64,
efa-1.14.2-1.amzn2.x86_64, dcv-2021.3.11591-1.el7.x86_64, slurm-21-08-6-1",
  "state": "AVAILABLE",
"tags": [
  {
    "value": "arn:aws:imagebuilder:us-east-1:123456789012:image/
parallelclusterimage-custom-alinux2-image/3.1.2/1",
    "key": "Ec2ImageBuilderArn"
  },
  {
    "value": "parallelcluster-1234abcd5678efgh-v1-do-not-delete",
    "key": "parallelcluster:s3_bucket"
  },
  {
    "value": "custom-alinux2-image",
    "key": "parallelcluster:image_name"
  },
  {
    "value": "available",
    "key": "parallelcluster:build_status"
  },
  {
    "value": "s3://parallelcluster-1234abcd5678efgh-v1-do-not-delete/
parallelcluster/3.1.2/images/custom-alinux2-image-1234abcd5678efgh/configs/image-
config.yaml",
    "key": "parallelcluster:build_config"
  },
  {
    "value": "EC2 Image Builder",
    "key": "CreatedBy"
  },
  {
    "value": "arn:aws:logs:us-east-1:123456789012:log-group:/aws/imagebuilder/
ParallelClusterImage-custom-alinux2-image",
    "key": "parallelcluster:build_log"
  }
]
}

```

```
  },
  {
    "value": "4.14.268-205.500.amzn2.x86_64",
    "key": "parallelcluster:kernel_version"
  },
  {
    "value": "arn:aws:imagebuilder:us-east-1:444455556666:image/amazon-linux-2-
x86/2022.3.16/1",
    "key": "parallelcluster:parent_image"
  },
  {
    "value": "3.1.2",
    "key": "parallelcluster:version"
  },
  {
    "value": "0.5.14",
    "key": "parallelcluster:munge_version"
  },
  {
    "value": "21-08-6-1",
    "key": "parallelcluster:slurm_version"
  },
  {
    "value": "2021.3.11591-1.el7.x86_64",
    "key": "parallelcluster:dcv_version"
  },
  {
    "value": "alinux2-image",
    "key": "parallelcluster:image_id"
  },
  {
    "value": "3.2.3",
    "key": "parallelcluster:pmix_version"
  },
  {
    "value": "parallelcluster/3.7.0/images/alinux2-image-abcd1234efgh56781234",
    "key": "parallelcluster:s3_image_dir"
  },
  {
    "value": "1.14.2-1.amzn2.x86_64",
    "key": "parallelcluster:efa_version"
  },
  {
    "value": "alinux2",
```

```
    "key": "parallelcluster:os"
  },
  {
    "value": "aws-parallelcluster-cookbook-3.1.2",
    "key": "parallelcluster:bootstrap_file"
  },
  {
    "value": "1.8.23-10.amzn2.1.x86_64",
    "key": "parallelcluster:sudo_version"
  },
  {
    "value": "2.10.8-5.amzn2.x86_64",
    "key": "parallelcluster:lustre_version"
  }
],
"architecture": "x86_64"
},
"version": "3.1.2"
}
```

pcluster export-cluster-logs

Exporte los registros del clúster a un archivo `tar.gz` local pasándolos por un bucket de Amazon S3.

```
pcluster export-cluster-logs [-h]
    --bucket BUCKET_NAME
    --cluster-name CLUSTER_NAME
  [--bucket-prefix BUCKET_PREFIX]
  [--debug]
  [--end-time END_TIME]
  [--filters FILTER [FILTER ...]]
  [--keep-s3-objects KEEP_S3_OBJECTS]
  [--output-file OUTPUT_FILE]
  [--region REGION]
  [--start-time START_TIME]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster export-cluster-logs`.

--bucket *BUCKET_NAME*

Especifica el nombre del bucket de Amazon S3 al que se exportarán los datos de los registros del clúster. Tiene que estar en la misma región que el clúster.

Note

Debe añadir permisos a la política de bucket de Amazon S3 para conceder el CloudWatch acceso. Para obtener más información, consulte [Establecer permisos en un bucket de Amazon S3](#) en la Guía del usuario de CloudWatch Logs.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--bucket-prefix *BUCKET_PREFIX*

Especifica la ruta en el bucket de Amazon S3 donde se almacenará la información de los registros exportados.

De forma predeterminada, el prefijo del bucket es:

```
cluster-name-logs-202209061743.tar.gz
```

202209061743 es la hora actual en formato %Y%m%d%H%M.

--debug

Habilita del registro de depuración.

--end-time *END_TIME*

Especifica el final del intervalo de tiempo para recopilar los eventos del registro, expresado en formato ISO 8601 (YYYY-MM-DDThh:mm:ssZ, por ejemplo 2021-01-01T20:00:00Z). No se incluyen los eventos con una marca de tiempo igual o posterior a esta hora. Se pueden omitir los elementos de tiempo (por ejemplo, minutos y segundos). El valor predeterminado es la hora actual.

--filters *FILTER [FILTER ...]*

Especifica los filtros para el registro. Formato: Name=a,Values=1 Name=b,Values=2,3. Los filtros compatibles son:

private-dns-name

Especifica la forma abreviada del nombre de DNS privado de la instancia (por ejemplo `ip-10-0-0-101`).

node-type

Especifica el tipo de nodo, el único valor aceptado para este filtro es `HeadNode`.

--keep-s3-objects *KEEP_S3_OBJECTS*

Si se establece en `true`, se conservan las exportaciones de objetos exportados a Amazon S3. (el valor predeterminado es `false`).

--output-file *OUTPUT_FILE*

Especifica la ruta del archivo en la que se guardará el archivo de registro. Si se proporciona, los registros se guardan localmente. De lo contrario, se cargan en Amazon S3 con la URL devuelta en la salida. De forma predeterminada, se cargan en Amazon S3.

--region, -r *REGION*

Especifica el Región de AWS que se va a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

--start-time *START_TIME*

Especifica el inicio del intervalo de tiempo, expresado en formato ISO 8601 (`YYYY-MM-DDThh:mm:ssZ`, por ejemplo `2021-01-01T20:00:00Z`). Se incluyen los eventos de registro con una marca de tiempo igual o posterior a esta hora. Si no se especifica, el valor predeterminado es la hora en que se creó el clúster.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster export-cluster-logs --bucket cluster-v3-bucket -n cluster-v3
{
  "url": "https://cluster-v3-bucket..."
}
```

pcluster export-image-logs

Exporte los registros de la pila del generador de imágenes en un archivo `tar.gz` local pasándolos por un bucket de Amazon S3.

```
pcluster export-image-logs [-h]
    --bucket BUCKET
    --image-id IMAGE_ID
    [--bucket-prefix BUCKET_PREFIX]
    [--debug]
    [--end-time END_TIME]
    [--keep-s3-objects KEEP_S3_OBJECTS]
    [--output-file OUTPUT_FILE]
    [--region REGION]
    [--start-time START_TIME]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster export-image-logs`.

--bucket *BUCKET_NAME*

Especifica el nombre del bucket de Amazon S3 al que se exportarán los registros de la creación de imágenes. Tiene que estar en la misma región que la imagen.

Note

Debe añadir permisos a la política de bucket de Amazon S3 para conceder el CloudWatch acceso. Para obtener más información, consulte [Establecer permisos en un bucket de Amazon S3](#) en la Guía del usuario de CloudWatch Logs.

--image-id, -i *IMAGE_ID*

El identificador de la imagen cuyos registros se exportarán.

--bucket-prefix *BUCKET_PREFIX*

Especifica la ruta en el bucket de Amazon S3 donde se almacenará la información de los registros exportados.

De forma predeterminada, el prefijo del bucket es:

```
ami-id-logs-202209061743.tar.gz
```

202209061743 es la hora actual en formato %Y%m%d%H%M.

--debug

Habilita del registro de depuración.

--end-time **END_TIME**

Especifica el final del intervalo de tiempo para recopilar los eventos del registro, expresado en formato ISO 8601 (YYYY-MM-DDThh:mm:ssZ, por ejemplo 2021-01-01T20:00:00Z). No se incluyen los eventos con una marca de tiempo igual o posterior a esta hora. Se pueden omitir los elementos de tiempo (por ejemplo, minutos y segundos). El valor predeterminado es la hora actual.

--keep-s3-objects **KEEP_S3_OBJECTS**

Si se establece en true, se conservan las exportaciones de objetos exportados a Amazon S3. (el valor predeterminado es false).

--output-file **OUTPUT_FILE**

Especifica la ruta del archivo en la que se guardará el archivo de registro. Si se proporciona, los registros se guardan localmente. De lo contrario, se cargan en Amazon S3 con la URL devuelta en la salida. De forma predeterminada, se cargan en Amazon S3.

--region, -r **REGION**

Especifica el Región de AWS que se va a utilizar. Región de AWS Debe especificarse mediante la variable de AWS_DEFAULT_REGION entorno, la region configuración de la [default] sección del ~/.aws/config archivo o el --region parámetro.

--start-time **START_TIME**

Especifica el inicio del intervalo de tiempo, expresado en formato ISO 8601 (YYYY-MM-DDThh:mm:ssZ, por ejemplo 2021-01-01T20:00:00Z). Se incluyen los eventos de registro con una marca de tiempo igual o posterior a esta hora. Si no se especifica, el valor predeterminado es la hora en que se creó el clúster.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster export-image-logs --bucket image-v3-bucket --image-id ami-1234abcd5678efgh
{
  "url": "https://image-v3-bucket..."
}
```

pcluster get-cluster-log-events

Recupere los eventos asociados a un flujo de registro.

```
pcluster get-cluster-log-events [-h]
    --cluster-name CLUSTER_NAME
    --log-stream-name LOG_STREAM_NAME
    [--debug]
    [--end-time END_TIME]
    [--limit LIMIT]
    [--next-token NEXT_TOKEN]
    [--query QUERY]
    [--region REGION]
    [--start-from-head START_FROM_HEAD]
    [--start-time START_TIME]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster get-cluster-log-events`.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--log-stream-name *LOG_STREAM_NAME*

Especifica el nombre del flujo de registro. Puede usar el comando `list-cluster-log-streams` para recuperar un flujo de registro asociado a uno o varios eventos.

--debug

Habilita del registro de depuración.

--end-time *END_TIME*

Especifica el final del intervalo de tiempo, expresado en formato ISO 8601 (YYYY-MM-DDThh:mm:ssZ, por ejemplo 2021-01-01T20:00:00Z). No se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

--limit *LIMIT*

Especifica el número máximo de eventos de registro devueltos. Si no se especifica nada, este valor será el número máximo de eventos de registro que quepan en un tamaño de respuesta de 1 MB, hasta 10 000 eventos de registro.

--next-token *NEXT_TOKEN*

Especifica el token que se va a utilizar para las solicitudes paginadas.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

--start-from-head *START_FROM_HEAD*

Si el valor es `true`, se devuelven primero los eventos de registro más antiguos. Si el valor es `false`, se devuelven primero los eventos de registro más recientes (el valor predeterminado es `false`).

--start-time *START_TIME*

Especifica el inicio del intervalo de tiempo, expresado en formato ISO 8601 (YYYY-MM-DDThh:mm:ssZ, por ejemplo `2021-01-01T20:00:00Z`). Se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster get-cluster-log-events \
  -c cluster-v3 \
  -r us-east-1 \
  --log-stream-name ip-198-51-100-44.i-1234567890abcdef0.clustermgtd \
  --limit 3
{
  "nextToken": "f/36966906399261933213029082268132291405859205452101451780/s",
  "prevToken": "b/36966906399239632467830551644990755687586557090595471362/s",
  "events": [
    {
      "message": "2022-07-12 19:16:53,379 - [slurm_plugin.clustermgtd:_maintain_nodes]
- INFO - Performing node maintenance actions",
      "timestamp": "2022-07-12T19:16:53.379Z"
    },
    {
```

```

    "message": "2022-07-12 19:16:53,380 - [slurm_plugin.clustermgtd:_maintain_nodes]
- INFO - Following nodes are currently in replacement: (x0) []",
    "timestamp": "2022-07-12T19:16:53.380Z"
  },
  {
    "message": "2022-07-12 19:16:53,380 -
[slurm_plugin.clustermgtd:_terminate_orphaned_instances] - INFO - Checking for
orphaned instance",
    "timestamp": "2022-07-12T19:16:53.380Z"
  }
]
}

```

pcluster get-cluster-stack-events

Recupera los eventos asociados a la pila del clúster especificado.

Note

A partir de la versión 3.6.0, AWS ParallelCluster utiliza pilas anidadas para crear los recursos asociados a las colas y los recursos de cómputo. La API de `GetClusterStackEvents` y el comando `pcluster get-cluster-stack-events` solo devuelven los eventos de la pila principal del clúster. Puede ver los eventos de la pila de clústeres, incluidos los relacionados con las colas y los recursos informáticos, en la consola. CloudFormation

```

pcluster get-cluster-stack-events [-h]
    --cluster-name CLUSTER_NAME
    [--debug]
    [--next-token NEXT_TOKEN]
    [--query QUERY]
    [--region REGION]

```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster get-cluster-stack-events`.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--debug

Habilita del registro de depuración.

--next-token *NEXT_TOKEN*

Especifica el token que se va a utilizar para las solicitudes paginadas.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster get-cluster-stack-events \  
  -n cluster-v3 \  
  -r us-east-1 \  
  --query "events[0]"  
{  
  "eventId": "1234abcd-56ef-78gh-90ij-abcd1234efgh",  
  "physicalResourceId": "arn:aws:cloudformation:us-east-1:123456789012:stack/cluster-  
v3/1234abcd-56ef-78gh-90ij-abcd1234efgh",  
  "resourceStatus": "CREATE_COMPLETE",  
  "stackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/cluster-  
v3/1234abcd-56ef-78gh-90ij-abcd1234efgh",  
  "stackName": "cluster-v3",  
  "logicalResourceId": "cluster-v3",  
  "resourceType": "AWS::CloudFormation::Stack",  
  "timestamp": "2022-07-12T18:29:12.140Z"  
}
```

pcluster get-image-log-events

Recupera los eventos asociados a la creación de una imagen.

```
pcluster get-image-log-events [-h]
```

```
--image-id IMAGE_ID
--log-stream-name LOG_STREAM_NAME
[--debug]
[--end-time END_TIME]
[--limit LIMIT]
[--next-token NEXT_TOKEN]
[--query QUERY]
[--region REGION]
[--start-from-head START_FROM_HEAD]
[--start-time START_TIME]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster get-image-log-events`.

--image-id, -i *IMAGE_ID*

Especifica el identificador de la imagen.

--log-stream-name *LOG_STREAM_NAME*

Especifica el nombre del flujo de registro. Puede usar el comando `list-image-log-streams` para recuperar un flujo de registro asociado a uno o varios eventos.

--debug

Habilita del registro de depuración.

--end-time *END_TIME*

Especifica el final del intervalo de tiempo, expresado en formato ISO 8601 (YYYY-MM-DDThh:mm:ssZ, por ejemplo 2021-01-01T20:00:00Z). No se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

--limit *LIMIT*

Especifica el número máximo de eventos de registro devueltos. Si no se especifica nada, este valor será el número máximo de eventos de registro que quepan en un tamaño de respuesta de 1 MB, hasta 10 000 eventos de registro.

--next-token *NEXT_TOKEN*

Especifica el token que se va a utilizar para las solicitudes paginadas.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

--start-from-head *START_FROM_HEAD*

Si el valor es `true`, se devuelven primero los eventos de registro más antiguos. Si el valor es `false`, se devuelven primero los eventos de registro más recientes (el valor predeterminado es `false`).

--start-time *START_TIME*

Especifica el inicio del intervalo de tiempo, expresado en formato ISO 8601 (YYYY-MM-DDThh:mm:ssZ, por ejemplo `2021-01-01T20:00:00Z`). Se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.2:

```
$ pcluster get-image-log-events --image-id custom-linux2-image --region us-east-1 --  
log-stream-name 3.1.2/1 --limit 3  
{  
  "nextToken": "f/36778317771100849897800729464621464113270312017760944178/s",  
  "prevToken": "b/36778317766952911290874033560295820514557716777648586800/s",  
  "events": [  
    {  
      "message": "ExecuteBash: FINISHED EXECUTION",  
      "timestamp": "2022-04-05T22:13:26.633Z"  
    },  
    {  
      "message": "Document arn:aws:imagebuilder:us-east-1:123456789012:component/  
parallelclusterimage-test-1234abcd-56ef-78gh-90ij-abcd1234efgh/3.1.2/1",  
      "timestamp": "2022-04-05T22:13:26.741Z"  
    },  
    {  
      "message": "TOE has completed execution successfully",  
      "timestamp": "2022-04-05T22:13:26.819Z"  
    }  
  ]  
}
```

```
]
}
```

pcluster get-image-stack-events

Recupera los eventos asociados a la pila para la creación de la imagen especificada.

```
pcluster get-image-stack-events [-h]
    --image-id IMAGE_ID
    [--debug]
    [--next-token NEXT_TOKEN]
    [--query QUERY]
    [--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster get-image-stack-events`.

--image-id, -i *IMAGE_ID*

Especifica el identificador de la imagen.

--debug

Habilita del registro de depuración.

--next-token *NEXT_TOKEN*

Especifica el token que se va a utilizar para las solicitudes paginadas.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.2:

```
$ pcluster get-image-stack-events --image-id custom-alinux2-image --region us-east-1 --
query "events[0]"
{
  "eventId": "ParallelClusterImage-CREATE_IN_PROGRESS-2022-04-05T21:39:24.725Z",
  "physicalResourceId": "arn:aws:imagebuilder:us-east-1:123456789012:image/
parallelclusterimage-custom-alinux2-image/3.1.2/1",
  "resourceStatus": "CREATE_IN_PROGRESS",
  "resourceStatusReason": "Resource creation Initiated",
  "resourceProperties": "{\"InfrastructureConfigurationArn\":
\\\"arn:aws:imagebuilder:us-east-1:123456789012:infrastructure-configuration/
parallelclusterimage-1234abcd-56ef-78gh-90ij-abcd1234efgh\\\",\\\"ImageRecipeArn
\\\":\\\"arn:aws:imagebuilder:us-east-1:123456789012:image-recipe/
parallelclusterimage-custom-alinux2-image/3.1.2\\\",\\\"DistributionConfigurationArn
\\\":\\\"arn:aws:imagebuilder:us-east-1:123456789012:distribution-
configuration/parallelclusterimage-1234abcd-56ef-78gh-90ij-abcd1234efgh\\\",
\\\"EnhancedImageMetadataEnabled\\\":\\\"false\\\",\\\"Tags\\\":{\\\"parallelcluster:image_name\\\":
\\\"custom-alinux2-image\\\",\\\"parallelcluster:image_id\\\":\\\"custom-alinux2-image\\\"}}\",
  "stackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/custom-alinux2-
image/1234abcd-56ef-78gh-90ij-abcd1234efgh",
  "stackName": "custom-alinux2-image",
  "logicalResourceId": "ParallelClusterImage",
  "resourceType": "AWS::ImageBuilder::Image",
  "timestamp": "2022-04-05T21:39:24.725Z"
}
```

pcluster list-clusters

Obtenga la lista de los clústeres existentes.

```
pcluster list-clusters [-h]
    [--cluster-status {CREATE_IN_PROGRESS,CREATE_FAILED,CREATE_COMPLETE,
DELETE_IN_PROGRESS,DELETE_FAILED,UPDATE_IN_PROGRESS,
UPDATE_COMPLETE,UPDATE_FAILED}]
    [{CREATE_IN_PROGRESS,CREATE_FAILED,CREATE_COMPLETE,
DELETE_IN_PROGRESS,DELETE_FAILED,UPDATE_IN_PROGRESS,
UPDATE_COMPLETE,UPDATE_FAILED} ...]]
    [--debug]
    [--next-token NEXT_TOKEN]
    [--query QUERY]
    [--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster list-clusters`.

--cluster-status {CREATE_IN_PROGRESS, CREATE_FAILED, CREATE_COMPLETE, DELETE_IN_PROGRESS, DELETE_FAILED, UPDATE_IN_PROGRESS, UPDATE_COMPLETE, UPDATE_FAILED} [{CREATE_IN_PROGRESS, CREATE_FAILED, CREATE_COMPLETE, DELETE_IN_PROGRESS, DELETE_FAILED, UPDATE_IN_PROGRESS, UPDATE_COMPLETE, UPDATE_FAILED} ...]

Especifica la lista de estados del clúster por los que se va a filtrar. (el valor predeterminado es `all`).

--debug

Habilita del registro de depuración.

--next-token *NEXT_TOKEN*

Especifica el token que se va a utilizar para las solicitudes paginadas.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster list-clusters
{
  "clusters": [
    {
      "clusterName": "cluster-v3",
      "cloudformationStackStatus": "CREATE_COMPLETE",
      "cloudformationStackArn": "arn:aws:cloudformation:us-east-1:123456789012:stack/cluster-v3/1234abcd-56ef-78gh-90ij-abcd1234efgh",
      "region": "us-east-1",
      "version": "3.1.4",
```



```
    "clusterStatus": "CREATE_COMPLETE"
  }
]
}
```

pcluster list-cluster-log-streams

Obtenga la lista de flujos de registro asociados a un clúster.

```
pcluster list-cluster-log-streams [-h]
    --cluster-name CLUSTER_NAME
    [--filters FILTERS [FILTERS ...]]
    [--next-token NEXT_TOKEN] [--debug]
    [--query QUERY]
    [--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster list-cluster-log-streams`.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--debug

Habilita el registro de depuración.

--filters *FILTERS* [*FILTERS* ...]

Especifica los filtros para los flujos de registro. Formato: `Name=a,Values=1`
`Name=b,Values=2,3`. Los filtros compatibles son:

`private-dns-name`

Especifica la forma abreviada del nombre de DNS privado de la instancia (por ejemplo
`ip-10-0-0-101`).

`node-type`

Especifica el tipo de nodo, el único valor aceptado para este filtro es `HeadNode`.

--next-token *NEXT_TOKEN*

Especifica el token que se va a utilizar para las solicitudes paginadas.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster list-cluster-log-streams \
  -n cluster-v3 \
  -r us-east-1 \
  --query 'logStreams[*].LogStreamName'
[
  "ip-172-31-58-205.i-1234567890abcdef0.cfn-init",
  "ip-172-31-58-205.i-1234567890abcdef0.chef-client",
  "ip-172-31-58-205.i-1234567890abcdef0.cloud-init",
  "ip-172-31-58-205.i-1234567890abcdef0.clustermgtd",
  "ip-172-31-58-205.i-1234567890abcdef0.slurmctld",
  "ip-172-31-58-205.i-1234567890abcdef0.supervisord",
  "ip-172-31-58-205.i-1234567890abcdef0.system-messages"
]
```

pcluster list-images

Obtenga la lista de imágenes personalizadas existentes.

```
pcluster list-images [-h]
  --image-status {AVAILABLE,PENDING,FAILED}
  [--debug]
  [--next-token NEXT_TOKEN]
  [--query QUERY]
  [--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster list-images`.

--image-status {AVAILABLE,PENDING,FAILED}

Filtre las imágenes devueltas según el estado proporcionado.

--debug

Habilita del registro de depuración.

--next-token *NEXT_TOKEN*

Especifica el token que se va a utilizar para las solicitudes paginadas.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.2:

```
$ pcluster list-images --image-status AVAILABLE
{
  "images": [
    {
      "imageId": "custom-alinux2-image",
      "imageBuildStatus": "BUILD_COMPLETE",
      "ec2AmiInfo": {
        "amiId": "ami-1234abcd5678efgh"
      },
      "region": "us-east-1",
      "version": "3.1.2"
    }
  ]
}
```

pcluster list-image-log-streams

Obtenga la lista de flujos de registro asociados a una imagen.

```
pcluster list-image-log-streams [-h]
```

```
--image-id IMAGE_ID
[--next-token NEXT_TOKEN] [--debug]
[--query QUERY]
[--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster list-image-log-streams`.

--image-id, -i *IMAGE_ID*

Especifica el identificador de la imagen.

--debug

Habilita del registro de depuración.

--next-token *NEXT_TOKEN*

Especifica el token que se va a utilizar para las solicitudes paginadas.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.2:

```
$ pcluster list-image-log-streams --image-id custom-alinux2-image --region us-east-1 --
query 'logStreams[*].logStreamName'
[
  "3.0.0/1",
  "3.1.2/1"
]
```

pcluster list-official-images

Describe las AWS ParallelCluster AMI oficiales.

```
pcluster list-official-images [-h]
    [--architecture ARCHITECTURE]
    [--debug]
    [--os OS]
    [--query QUERY]
    [--region REGION]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster list-official-images`.

--architecture ARCHITECTURE

Especifica la arquitectura que se utilizará para filtrar los resultados. Si no se especifica este parámetro, se devuelven todas las arquitecturas.

--debug

Habilita del registro de depuración.

--os OS

Especifica el sistema operativo que se utilizará para filtrar los resultados. Si no se especifica este parámetro, se devuelven todos los sistemas operativos.

--query QUERY

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r REGION

Especifica las Región de AWS que se van a utilizar. Región de AWS Debe especificarse mediante el ajuste [Región](#) del archivo de configuración de la imagen, la variable de `AWS_DEFAULT_REGION` entorno, el `region` ajuste de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.2:

```
$ pcluster list-official-images
{
  "images": [
    {
      "amiId": "ami-015cf4e0d6306b2",
```

```
    "os": "ubuntu2004",
    "name": "aws-parallelcluster-3.1.2-ubuntu-2004-lts-hvm-x86_64-202202261505
2022-02-26T15-08-34.759Z",
    "version": "3.1.2",
    "architecture": "x86_64"
  },
  {
    "amiId": "ami-036f23237ce49d25b",
    "os": "ubuntu2204",
    "name": "aws-parallelcluster-3.1.2-ubuntu-1804-lts-hvm-x86_64-202202261505
2022-02-26T15-08-17.558Z",
    "version": "3.1.2",
    "architecture": "x86_64"
  },
  {
    "amiId": "ami-09e5327e694d89ef4",
    "os": "ubuntu2004",
    "name": "aws-parallelcluster-3.1.2-ubuntu-2004-lts-hvm-arm64-202202261505
2022-02-26T15-08-45.736Z",
    "version": "3.1.2",
    "architecture": "arm64"
  },
  {
    "amiId": "ami-0b9b0874c35f626ae",
    "os": "alinux2",
    "name": "aws-parallelcluster-3.1.2-amzn2-hvm-x86_64-202202261505
2022-02-26T15-08-31.311Z",
    "version": "3.1.2",
    "architecture": "x86_64"
  },
  {
    "amiId": "ami-0bf6d01f398f3737e",
    "os": "centos7",
    "name": "aws-parallelcluster-3.1.2-centos7-hvm-x86_64-202202261505
2022-02-26T15-08-25.001Z",
    "version": "3.1.2",
    "architecture": "x86_64"
  },
  {
    "amiId": "ami-0d0de4f95f56374bc",
    "os": "alinux2",
    "name": "aws-parallelcluster-3.1.2-amzn2-hvm-arm64-202202261505
2022-02-26T15-08-46.088Z",
    "version": "3.1.2",
```

```

    "architecture": "arm64"
  },
  {
    "amiId": "ami-0ebf7bc54b8740dc6",
    "os": "ubuntu2204",
    "name": "aws-parallelcluster-3.1.2-ubuntu-1804-lts-hvm-arm64-202202261505
2022-02-26T15-08-45.293Z",
    "version": "3.1.2",
    "architecture": "arm64"
  }
]
}

```

pcluster ssh

Ejecuta un comando ssh con el nombre de usuario y la dirección IP del clúster completados previamente. Se adjuntan argumentos arbitrarios al final de la línea de comandos ssh.

```

pcluster ssh [-h]
              --cluster-name CLUSTER_NAME
              [--debug]
              [--dryrun DRYRUN]
              [--region REGION]

```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster ssh`.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster al que hay que conectarse.

--debug

Habilita del registro de depuración.

--dryrun *DRYRUN*

Cuando se establece en `true`, muestra la línea de comandos que se va a ejecutar y se cierra (el valor predeterminado es `false`).

--region, -r *REGION*

Especifica el que se va Región de AWS a utilizar. Región de AWS Debe especificarse mediante la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

Ejemplo:

```
$ pcluster ssh --cluster-name mycluster -i ~/.ssh/id_rsa
```

Ejecuta un comando `ssh` con el nombre de usuario y la dirección IP del clúster completados previamente:

```
ssh ec2-user@1.1.1.1 -i ~/.ssh/id_rsa
```

pcluster update-cluster

Actualiza un clúster existente para que coincida con los ajustes de un archivo de configuración especificado.

```
pcluster update-cluster [-h]
                        --cluster-configuration CLUSTER_CONFIGURATION
                        --cluster-name CLUSTER_NAME
                        [--debug]
                        [--dryrun DRYRUN]
                        [--force-update FORCE_UPDATE]
                        [--query QUERY]
                        [--region REGION]
                        [--suppress-validators SUPPRESS_VALIDATORS [SUPPRESS_VALIDATORS ...]]
                        [--validation-failure-level {INFO,WARNING,ERROR}]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster update-cluster`.

--cluster-configuration, -c *CLUSTER_CONFIGURATION*

Especifica el archivo YAML de configuración del clúster.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--debug

Habilita del registro de depuración.

--dryrun *DRYRUN*

Cuando se establece en `true`, realiza la validación sin actualizar el clúster ni crear ningún recurso. Se puede utilizar para validar la configuración de la imagen y los requisitos de actualización (el valor predeterminado es `false`).

--force-update *FORCE_UPDATE*

Cuando se establece en `true`, fuerza la actualización ignorando los errores de validación de la actualización (el valor predeterminado es `false`).

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Especifica el Región de AWS que se va a utilizar. Región de AWS Debe especificarse mediante la [Region](#) configuración del archivo de configuración del clúster, la variable de `AWS_DEFAULT_REGION` entorno, la `region` configuración de la `[default]` sección del `~/.aws/config` archivo o el `--region` parámetro.

--suppress-validators *SUPPRESS_VALIDATORS* [*SUPPRESS_VALIDATORS ...*]

Identifica uno o más validadores de configuración para suprimirlos.

Formato: `(ALL|type:[A-Za-z0-9]+)`

--validation-failure-level *{INFO,WARNING,ERROR}*

Especifica el nivel de errores de validación notificados para la actualización.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster update-cluster -c cluster-config.yaml -n cluster-v3 -r us-east-1
{
  "cluster": {
    "clusterName": "cluster-v3",
    "cloudformationStackStatus": "UPDATE_IN_PROGRESS",
```

```

    "cloudformationStackArn": "arn:aws:cloudformation:us-east-1:123456789012:stack/
cluster-v3/1234abcd-56ef-78gh-90ij-abcd1234efgh",
    "region": "us-east-1",
    "version": "3.1.4",
    "clusterStatus": "UPDATE_IN_PROGRESS"
  },
  "changeSet": [
    {
      "parameter": "HeadNode.Iam.S3Access",
      "requestedValue": {
        "BucketName": "pc-beta-test",
        "KeyName": "output",
        "EnableWriteAccess": false
      }
    },
    {
      "parameter": "HeadNode.Iam.S3Access",
      "currentValue": {
        "BucketName": "pcluster-east-test-bucket",
        "KeyName": "logs",
        "EnableWriteAccess": true
      }
    }
  ]
}

```

pcluster update-compute-fleet

Actualiza el estado de la flota de computación del clúster.

```

pcluster update-compute-fleet [-h]
    --cluster-name CLUSTER_NAME
    --status {START_REQUESTED,STOP_REQUESTED,ENABLED,DISABLED}

    [--debug]
    [--query QUERY]
    [--region REGION]

```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de `pcluster update-compute-fleet`.

--cluster-name, -n *CLUSTER_NAME*

Especifica el nombre del clúster.

--status {START_REQUESTED,STOP_REQUESTED,ENABLED,DISABLED}

Especifica el estado aplicado a la flota de computación del clúster. Los estados START_REQUESTED y STOP_REQUESTED corresponden al planificador de slurm, mientras que los estados ENABLED y corresponden al planificador. DISABLED AWS Batch

--debug

Habilita del registro de depuración.

--query *QUERY*

Especifica la consulta JMESPath que se va a realizar en la salida.

--region, -r *REGION*

Región de AWS Especifica el que se va a utilizar. Región de AWS Debe especificarse mediante la variable de AWS_DEFAULT_REGION entorno, la region configuración de la [default] sección del ~/.aws/config archivo o el --region parámetro.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster update-compute-fleet -n cluster-v3 --status STOP_REQUESTED
{
  "status": "STOP_REQUESTED",
  "lastStatusUpdateTime": "2022-07-12T20:19:47.653Z"
}
```

pcluster version

Muestra la versión de. AWS ParallelCluster

```
pcluster version [-h] [--debug]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de pcluster version.

--debug

Habilita del registro de depuración.

Ejemplo que utiliza AWS ParallelCluster la versión 3.1.4:

```
$ pcluster version
{
  "version": "3.1.4"
}
```

pcluster3-config-converter

Lee un archivo de configuración de AWS ParallelCluster versión 2 y escribe un archivo de configuración de AWS ParallelCluster versión 3.

```
pcluster3-config-converter [-h]
                          [-t CLUSTER_TEMPLATE]
                          [-c CONFIG_FILE]
                          [--force-convert]
                          [-o OUTPUT_FILE]
```

Argumentos con nombre

-h, --help

Muestra el texto de ayuda de pcluster3-config-converter.

-t *CLUSTER_TEMPLATE*, --cluster-template *CLUSTER_TEMPLATE*

Especifica la [sección de \[cluster\]](#) del archivo de configuración que se va a convertir. Si no se especifica, el script buscará el parámetro [cluster-template](#) en la [sección de \[global\]](#) o buscará [cluster default].

-c *CONFIG_FILE*, --config-file *CONFIG_FILE*


Especifica el archivo de configuración de AWS ParallelCluster versión 2 que se va a leer.

--force-convert

Permite una conversión incluso si una o más configuraciones no son compatibles y no se recomiendan.

-o *OUTPUT_FILE*, --output-file *OUTPUT_FILE*

Especifica el archivo de configuración de AWS ParallelCluster versión 3 que se va a escribir. Si no se especifica este parámetro, la configuración se escribe como stdout.

 Note

El comando `pcluster3-config-convert` se agregó en AWS ParallelCluster versión 3.0.1.

Archivos de configuración

AWS ParallelCluster usa archivos YAML 1.1 para los parámetros de configuración.

Temas

- [Configuración del clúster](#)
- [Cree archivos de configuración de imágenes](#)

Configuración del clúster

AWS ParallelCluster la versión 3 utiliza archivos de configuración independientes para controlar la definición de la infraestructura de clústeres y la definición de las AMI personalizadas. Todos los archivos de configuración utilizan archivos YAML 1.1. La información detallada de cada uno de estos archivos de configuración se encuentra en el siguiente enlace. Para ver algunos ejemplos de configuraciones, consulte https://github.com/aws/aws-parallelcluster/tree/release-3.0/cli/tests/pcluster/example_configs.

Estos objetos se utilizan para la configuración del clúster de la AWS ParallelCluster versión 3.

Temas

- [Propiedades del fichero de configuración del clúster](#)
- [Sección de Imds](#)
- [Sección de Image](#)
- [Sección de HeadNode](#)
- [Sección de Scheduling](#)

- [Sección de SharedStorage](#)
- [Sección de iam](#)
- [Sección de LoginNodes](#)
- [Sección de Monitoring](#)
- [Sección de Tags](#)
- [Sección de AdditionalPackages](#)
- [Sección de DirectoryService](#)
- [Sección de DeploymentSettings](#)

Propiedades del fichero de configuración del clúster

Region (Opcional, String)

Especifica el Región de AWS para el clúster. Por ejemplo, `us-east-2`.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

CustomS3Bucket (Opcional, String)

Especifica el nombre de un bucket de Amazon S3 que se crea en Cuenta de AWS usted para almacenar los recursos que utilizan sus clústeres, como el archivo de configuración del clúster. AWS ParallelCluster mantiene un bucket de Amazon S3 en cada uno de los sitios en los Región de AWS que cree los clústeres. De forma predeterminada, estos buckets de Amazon S3 se llaman `parallelcluster-hash-v1-DO-NOT-DELETE`.

[Política de actualización: si se cambia esta configuración, no se permite la actualización. Si fuerzas la actualización, se ignorará el nuevo valor y se utilizará el anterior.](#)

AdditionalResources (Opcional, String)

Define una AWS CloudFormation plantilla adicional para lanzarla junto con el clúster. Esta plantilla adicional se utiliza para la creación de recursos que existen fuera del clúster pero que forman parte del ciclo de vida de este.

Este valor tiene que ser una dirección URL HTTP a una plantilla pública con todos los parámetros incluidos.

No hay valor predeterminado.

Política de actualización: esta configuración se puede cambiar durante una actualización.

Sección de **Imds**

(Opcional) Especifica la configuración del servicio de metadatos de instancias (IMDS) global.

```
Imds:  
ImdsSupport: string
```

Propiedades de **Imds**

ImdsSupport (Opcional, String)

Especifica qué versiones de IMDS se admiten en los nodos del clúster. Los valores admitidos son `v1.0` y `v2.0`. El valor predeterminado es `v2.0`.

Si `ImdsSupport` se establece en `v1.0`, se admiten tanto IMDSv1 como IMDSv2.

Si `ImdsSupport` se establece en `v2.0`, solo se admite IMDSv2.

Para obtener más información, consulte [Utilizar IMDSv2](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Note

A partir de la AWS ParallelCluster versión 3.7.0, el valor `ImdsSupport` predeterminado es `v2.0`. Le recomendamos que configure `ImdsSupport` en `v2.0` y sustituya IMDSv1 por IMDSv2 en sus llamadas a acciones personalizadas.

El soporte para [Imds/ImdsSupport](#) se añade con la AWS ParallelCluster versión 3.3.0.

Sección de **Image**

(Obligatorio) Define el sistema operativo del clúster.

```
Image:  
Os: string
```

`CustomAmi`: *string*

Propiedades de **Image**

`Os` (Requerido, *String*)

Especifica el sistema operativo que se va a utilizar en el clúster. Los valores admitidos son `alinux2`, `centos7`, `ubuntu2204`, `ubuntu2004`, `rhel8`, `rocky8rhel9`, `rocky9`.

Note

RedHat Enterprise Linux 8.7 (`rhel8`) se agrega a partir de la AWS ParallelCluster versión 3.6.0.

Si configura el clúster para usar `rhel`, el costo bajo demanda de cualquier tipo de instancia es mayor que si configura el clúster para usar otros sistemas operativos compatibles. Para obtener más información sobre los precios, consulte [Precios bajo demanda](#) y [¿Cómo se ofrece y se cotiza Red Hat Enterprise Linux en Amazon EC2?](#).

RedHat Enterprise Linux 9 (`rhel9`) se agrega a partir de la versión 3.9.0. AWS ParallelCluster

Aparte de los específicos Regiones de AWS mencionados en la siguiente tabla que no son compatibles. `centos7` Todas las demás regiones AWS comerciales son compatibles con los siguientes sistemas operativos.

Partición (Regiones de AWS)	<code>alinux2</code>	<code>centos7</code>	<code>ubuntu2204</code> y <code>ubuntu2004</code>	<code>rhel8</code>	<code>rhel9</code>
Comercial (Regiones de AWS ninguno de ellos se menciona específicamente)	True	True	True	True	True
AWS GovCloud (Este de EE. UU.) (<code>us-gov-east-1</code>)	True	False	True	True	True

Partición (Regiones de AWS)	alinux2	centos7	ubuntu2204 y ubuntu2004	rhel8	rhel9
AWS GovCloud (EEUU-Oeste) (us-gov-west-1)	True	False	True	True	True
China (Pekín) (cn-north-1)	True	False	True	True	True
China (Ningxia) (cn-northwest-1)	True	False	True	True	True

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Note

AWS ParallelCluster La versión 3.8.0 es compatible con Rocky Linux 8, pero las AMI Rocky Linux 8 prediseñadas (para arquitecturas x86 y ARM) no están disponibles. AWS ParallelCluster La versión 3.8.0 permite crear clústeres con Rocky Linux 8 mediante AMI personalizadas. Para obtener más información, consulte [Consideraciones sobre el sistema operativo](#) AWS ParallelCluster La versión 3.9.0 es compatible con Rocky Linux 9, pero las AMI Rocky Linux 9 prediseñadas (para arquitecturas x86 y ARM) no están disponibles. AWS ParallelCluster La versión 3.9.0 admite la creación de clústeres con Rocky Linux 9 mediante AMI personalizadas. Para obtener más información, consulte [Consideraciones sobre el sistema operativo](#).

CustomAmi (Opcional, String)

Especifica el ID de una AMI personalizada que se va a usar para los nodos principal y de computación en lugar de la AMI publicada predeterminada. Para obtener más información, consulte [AWS ParallelCluster Personalización de AMI](#).

Si la AMI personalizada requiere permisos adicionales para su lanzamiento, estos permisos se deben agregar a las políticas de usuario y de nodo principal.

Por ejemplo, si una AMI personalizada tiene asociada una instantánea cifrada, se requieren las siguientes políticas adicionales tanto en las políticas de usuario como en las del nodo principal:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:DescribeKey",
        "kms:ReEncrypt*",
        "kms:CreateGrant",
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:<AWS_REGION>:<AWS_ACCOUNT_ID>;key/<AWS_KMS_KEY_ID>"
      ]
    }
  ]
}
```

Para crear una AMI personalizada de RedHat Enterprise Linux, debe configurar el sistema operativo para instalar los paquetes que proporcionan los repositorios RHUI (AWS): `rhel-
<version>-baseos-rhui-rpmsrhel-
<version>-appstream-rhui-rpms`, y `codeready-builder-for-rhel-
<version>-rhui-rpms`. Además, los repositorios de la AMI personalizada deben contener paquetes de `kernel-devel` de la misma versión que la versión del núcleo en ejecución. `kernel`.

Limitaciones conocidas

- Solo RHEL 8.2 y las versiones posteriores admiten FSx para Lustre.
- La versión 4.18.0-425.3.1.el8 del kernel de RHEL 8.7 no admite FSx para Lustre.
- Solo RHEL 8.4 y las versiones posteriores son compatibles con EFA.

Para solucionar problemas de las advertencias de validación de AMI personalizadas, consulte [Solución de problemas con las AMI de](#) .

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Sección de **HeadNode**

(Obligatorio) Especifica la configuración del nodo principal.

```
HeadNode:
  InstanceType: string
  Networking:
    SubnetId: string
    ElasticIp: string/boolean
    SecurityGroups:
      - string
    AdditionalSecurityGroups:
      - string
  Proxy:
    HttpProxyAddress: string
  DisableSimultaneousMultithreading: boolean
  Ssh:
    KeyName: string
    AllowedIps: string
  LocalStorage:
    RootVolume:
      Size: integer
      Encrypted: boolean
      VolumeType: string
      Iops: integer
      Throughput: integer
      DeleteOnTermination: boolean
    EphemeralVolume:
      MountDir: string
  SharedStorageType: string
  Dcv:
    Enabled: boolean
    Port: integer
    AllowedIps: string
  CustomActions:
    OnNodeStart:
      Sequence:
        - Script: string
          Args:
            - string
          Script: string
          Args:
            - string
    OnNodeConfigured:
```

```
Sequence:
  - Script: string
    Args:
      - string
  Script: string
  Args:
    - string
OnNodeUpdated:
  Sequence:
    - Script: string
      Args:
        - string
    Script: string
    Args:
      - string
Iam:
  InstanceRole: string
  InstanceProfile: string
  S3Access:
    - BucketName: string
      EnableWriteAccess: boolean
      KeyName: string
  AdditionalIamPolicies:
    - Policy: string
Imds:
  Secured: boolean
Image:
  CustomAmi: string
```

Propiedades de **HeadNode**

InstanceType (Requerido, String)

Especifica el tipo de instancia para el nodo principal.

Especifica el tipo de instancia de Amazon EC2 que se utiliza para el nodo principal. La arquitectura del tipo de instancia debe ser la misma que la utilizada para la configuración AWS Batch [InstanceType](#)o Slurm [InstanceType](#).

Note

AWS ParallelCluster no admite los siguientes tipos de instancias para la HeadNode configuración.

- `hpc6id`

Si define un tipo de instancia `p4d` u otro tipo de instancia que tenga varias interfaces de red o una tarjeta de interfaz de red para el nodo principal, debe configurar `ElasticIp/true/` en para proporcionar acceso público. Las IP públicas de solo se pueden asignar a instancias lanzadas con una única interfaz de red. En este caso, le recomendamos que utilice una [puerta de enlace NAT](#) para proporcionar acceso público a los nodos de computación del clúster. Para obtener más información, consulte [Asignación de una dirección IPv4 pública durante el lanzamiento de una instancia](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

`DisableSimultaneousMultithreading` (Opcional, Boolean)

Si es `true`, desactiva el hiperprocesamiento en el nodo principal. El valor predeterminado es `false`.

No todos los tipos de instancias pueden deshabilitar la tecnología Hyper-Threading. Para obtener una lista de los tipos de instancias que admiten la desactivación del hiperproceso, consulte [Núcleos y subprocesos de CPU para cada núcleo de CPU por tipo de instancia](#) en la Guía del usuario de Amazon EC2.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

`SharedStorageType` (Opcional, String)

Especifica el tipo de almacenamiento que se utiliza para los datos compartidos internamente. Los datos compartidos internamente incluyen los datos que se AWS ParallelCluster utilizan para administrar el clúster y los que se comparten por defecto, `/home` si no se especifican en el [Sección de SharedStorage](#) directorio de montaje, para montar un volumen de sistema de archivos compartido. Para obtener más información sobre los datos compartidos internos, consulte. [AWS ParallelClusterDirectorios internos](#)

Si `Ebs` es el tipo de almacenamiento predeterminado, el nodo principal exportará partes de su volumen raíz como directorios compartidos para los nodos de cómputo y los nodos de inicio de sesión mediante NFS.

Si `Efs`, Parallelcluster creará un sistema de archivos EFS para usarlo con datos internos compartidos y `/home`

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Note

Cuando el clúster se amplía, el tipo de almacenamiento de EBS puede presentar cuellos de botella en el rendimiento, ya que el nodo principal comparte los datos del volumen raíz con los nodos de procesamiento mediante exportaciones a NFS. Con EFS, puede evitar las exportaciones de NFS a medida que el clúster se amplía y evitar los cuellos de botella de rendimiento asociados a ellas. Se recomienda elegir EBS para aprovechar al máximo el potencial de lectura/escritura de archivos pequeños y del proceso de instalación. Elija EFS para escalar.

Networking

(Obligatorio) Define la configuración de red del nodo principal.

Networking:

SubnetId: *string*

ElasticIp: *string/boolean*

SecurityGroups:

- *string*

AdditionalSecurityGroups:

- *string*

Proxy:

HttpProxyAddress: *string*

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Propiedades de **Networking**

SubnetId (Requerido, String)

Especifica el ID de una subred existente en la que se va a aprovisionar el nodo maestro.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

ElasticIp (Opcional, String)

Crea o asigna una dirección IP elástica al nodo principal. Los valores admitidos son `true`, `false` o el ID de una dirección IP elástica existente. El valor predeterminado es `false`.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

SecurityGroups (Opcional, [String])

Lista de identificadores de grupos de seguridad de Amazon VPC que se van a utilizar en el nodo principal. Estos sustituyen a los grupos de seguridad que se AWS ParallelCluster crean si no se incluye esta propiedad.

Compruebe que los grupos de seguridad estén configurados correctamente para sus [SharedStorage](#) sistemas.

Política de actualización: esta configuración se puede cambiar durante una actualización.

AdditionalSecurityGroups (Opcional, [String])

Lista de identificadores de grupos de seguridad de Amazon VPC adicionales que se van a utilizar en el nodo principal.

Política de actualización: esta configuración se puede cambiar durante una actualización.

Proxy (Opcional)

Especifica la configuración del proxy para el nodo principal.

```
Proxy:  
  HttpProxyAddress: string
```

HttpProxyAddress (Opcional, String)

Define un servidor proxy HTTP o HTTPS, normalmente `https://x.x.x.x:8080`.

No hay valor predeterminado.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Ssh

(Opcional) Define la configuración para el acceso SSH al nodo principal.

```
Ssh:  
  KeyName: string  
  AllowedIps: string
```

Política de actualización: esta configuración se puede cambiar durante una actualización.

Propiedades de **Ssh**

KeyName (Opcional, String)

Nombra un par de claves de Amazon EC2 existente para habilitar el acceso a través de SSH al nodo principal.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

AllowedIps (Opcional, String)

Especifica el rango de IP con formato CIDR o un identificador de lista de prefijos para las conexiones SSH al nodo principal. El valor predeterminado es `0.0.0.0/0`.

Política de actualización: esta configuración se puede cambiar durante una actualización.

LocalStorage

(Opcional) Define la configuración de almacenamiento local para el nodo principal.

```
LocalStorage:  
  RootVolume:  
    Size: integer  
    Encrypted: boolean  
    VolumeType: string  
    Iops: integer  
    Throughput: integer  
    DeleteOnTermination: boolean  
  EphemeralVolume:  
    MountDir: string
```

Política de actualización: esta configuración se puede cambiar durante una actualización.

Propiedades de **LocalStorage**

RootVolume (Obligatorio)

Especifica el volumen de almacenamiento raíz del nodo principal.

```
RootVolume:  
  Size: integer  
  Encrypted: boolean  
  VolumeType: string
```


`Iops`: *integer*
`Throughput`: *integer*
`DeleteOnTermination`: *boolean*

Política de actualización: esta configuración se puede cambiar durante una actualización.

Size (Opcional, Integer)

Especifica el tamaño del volumen raíz del nodo principal en gibibytes (GiB). El tamaño predeterminado proviene de la AMI. El uso de un tamaño diferente requiere que la AMI admita growroot.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Encrypted (Opcional, Boolean)

Especifica si el volumen de está cifrado. El valor predeterminado es true.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

VolumeType (Opcional, String)

Especifica el tipo de [volumen de Amazon EBS](#). Los valores admitidos son gp2, gp3, io1, io2, sc1, st1 y standard. El valor predeterminado es gp3.

Para obtener más información, consulte [Tipos de volúmenes de Amazon EBS](#) en la Guía del usuario de Amazon EC2.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Iops (Opcional, Integer)

Define el número de IOPS para volúmenes de tipo io1, io2 y gp3.

El valor predeterminado, los valores admitidos y la proporción de datos volume_iops a volume_size varía entre VolumeType y Size.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

VolumeType = io1

Iops predeterminado = 100

Valores admitidos Iops = 100–64 000 †

Relación máxima Iops a Size = 50 IOPS por GiB. 5000 IOPS requieren un Size de al menos 100 GiB.

VolumeType = io2

Iops predeterminado = 100

Valores admitidos Iops = 100–64 000 (256 000 para los volúmenes de io2 Block Express) †

Relación máxima Iops a Size = 500 IOPS por GiB. 5000 IOPS requieren un Size de al menos 10 GiB.

VolumeType = gp3

Iops predeterminado = 3000

Valores admitidos Iops = 3000–16 000

Relación máxima Iops a Size = 500 IOPS por GiB. 5000 IOPS requieren un Size de al menos 10 GiB.

† Las IOPS máximas solo se garantizan en [las instancias creadas en el sistema Nitro](#) provisionadas con más de 32 000 IOPS. Otras instancias garantizan hasta 32,000 IOPS. Es posible que los volúmenes io1 más antiguos no alcancen el rendimiento máximo a menos que [modifique el volumen](#). io2 Los volúmenes de Block Express admiten valores de Iops de hasta 256 000 en tipos de instancias R5b. Para obtener más información, consulte [io2Block Express Volumes](#) en la Guía del usuario de Amazon EC2.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

Throughput (Opcional, Integer)

Define el rendimiento de los tipos de volumen gp3, en MiB/s. Esta configuración solo es válida cuando VolumeType se establece en gp3. El valor predeterminado es 125. Valores admitidos = 125–1000 MiB/s

La relación entre Throughput y Iops no puede ser superior a 0,25. El rendimiento máximo de 1000 MiB/s requiere que la configuración de Iops sea de al menos 4000.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

DeleteOnTermination (Opcional, Boolean)

Especifica si se debe eliminar el volumen raíz al finalizar el nodo principal. El valor predeterminado es true.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

EphemeralVolume (Opcional)

Especifica los detalles de cualquier volumen de almacén de instancias. Para obtener más información, consulte [Volúmenes de almacén de instancias](#) en la Guía del usuario de Amazon EC2.

EphemeralVolume:
MountDir: *string*

Política de actualización: si se cambia esta configuración, no se permite la actualización.

MountDir (Opcional, String)

Especifica el directorio de montaje del volumen del almacén de instancias. El valor predeterminado es `/scratch`.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Dcv

Define los valores de configuración del servidor de NICE DCV que se ejecuta en el nodo principal.

Para obtener más información, consulte [Conexión al nodo principal a través de NICE DCV](#).

Dcv:
Enabled: *boolean*
Port: *integer*
AllowedIps: *string*

Important

De forma predeterminada, el puerto NICE DCV configurado por AWS ParallelCluster está abierto a todas las direcciones IPv4. Sin embargo, los usuarios solo pueden conectarse a un puerto de NICE DCV si tienen la dirección URL de la sesión de NICE DCV y se conectan a la sesión de NICE DCV en un plazo de 30 segundos a partir del momento en que se devuelve la dirección URL de `pcluster dcv-connect`. Utilice el valor `AllowedIps` para restringir aún más el acceso al puerto de NICE DCV con un intervalo de direcciones IP con formato de CIDR y utilice la configuración de `Port` para establecer un puerto no estándar.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Propiedades de **Dcv**

Enabled (Requerido, Boolean)

Especifica si NICE DCV está habilitado en el nodo principal. El valor predeterminado es `false`.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Note

NICE DCV genera automáticamente un certificado autofirmado que se utiliza para proteger el tráfico entre el cliente de NICE DCV y el servidor NICE DCV en el nodo principal. Para configurar su propio certificado, consulte [Certificado HTTPS NICE DCV](#).

Port (Opcional, Integer)

(Opcional) Especifica el puerto de NICE DCV. El valor predeterminado es 8443.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

AllowedIps (opcional, recomendado, String)

(Opcional) Especifica el intervalo de direcciones IP con formato CIDR para las conexiones a NICE DCV. Esta configuración solo se usa cuando se AWS ParallelCluster crea el grupo de seguridad. El valor predeterminado es `0.0.0.0/0`, que permite el acceso desde cualquier dirección de Internet.

Política de actualización: esta configuración se puede cambiar durante una actualización.

CustomActions

(Opcional) Especifica los scripts personalizados que se ejecutarán en el nodo principal.

```
CustomActions:  
  OnNodeStart:  
    Sequence:  
      - Script: string  
        Args:  
          - string  
    Script: string
```

Args:

- *string*

OnNodeConfigured:Sequence:

- Script: *string*

Args:

- *string*

Script: *string*Args:

- *string*

OnNodeUpdated:Sequence:

- Script: *string*

Args:

- *string*

Script: *string*Args:

- *string*

Propiedades de **CustomActions**

OnNodeStart (Opcional)

Especifica un único script o una secuencia de scripts que se ejecutarán en el nodo principal antes de que se inicie cualquier acción de arranque de implementación del nodo. Para obtener más información, consulte [Acciones de arranque personalizadas](#).

Sequence (opcional)

Lista de scripts que se van a ejecutar. AWS ParallelCluster ejecuta los scripts en el mismo orden en que aparecen en el archivo de configuración, empezando por el primero.

Script (Requerido, String)

Especifica el archivo que se va a encapsular. La ruta del archivo debe comenzar con `https://` o `s3://`.

Args (Opcional, [String])

La lista de argumentos que se pasan al script único.

Script (Requerido, String)

Especifica el archivo que se va a utilizar para un único script. La ruta del archivo debe comenzar con `https://` o `s3://`.

Args (Opcional, [String])

La lista de argumentos que se pasan al script único.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

OnNodeConfigured (Opcional)

Especifica un único script o una secuencia de scripts que se ejecutarán en el nodo principal una vez finalizadas las acciones de arranque del nodo. Para obtener más información, consulte [Acciones de arranque personalizadas](#).

Sequence (Opcional)

Especifica la lista de scripts que se van a ejecutar.

Script (Requerido, String)

Especifica el archivo que se va a encapsular. La ruta del archivo debe comenzar con `https://` o `s3://`.

Args (Opcional, [String])

La lista de argumentos que se pasan al script único.

Script (Requerido, String)

Especifica el archivo que se va a utilizar para un único script. La ruta del archivo debe comenzar con `https://` o `s3://`.

Args (Opcional, [String])

La lista de argumentos que se pasan al script único.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

OnNodeUpdated (Opcional)

Especifica un único script o una secuencia de scripts que se ejecutarán en el nodo principal una vez finalizadas las acciones de actualización del nodo. Para obtener más información, consulte [Acciones de arranque personalizadas](#).

Sequence (Opcional)

Especifica la lista de scripts que se van a ejecutar.

Script (Requerido, String)

Especifica el archivo que se va a encapsular. La ruta del archivo debe comenzar con `https://` o `s3://`.

Args (Opcional, [String])

La lista de argumentos que se pasan al script único.

Script (Requerido, String)

Especifica el archivo que se va a utilizar para el único script. La ruta del archivo debe comenzar con `https://` o `s3://`.

Args (Opcional, [String])

La lista de argumentos que se pasan al script único.

Política de actualización: esta configuración se puede cambiar durante una actualización.

Note

`OnNodeUpdated` se añade a partir de la AWS ParallelCluster versión 3.4.0.
`Sequence` se añade a partir de la AWS ParallelCluster versión 3.6.0. Si lo especifica `Sequence`, puede enumerar varios scripts para una acción personalizada. AWS ParallelCluster sigue admitiendo la configuración de una acción personalizada con un único script, sin incluirlo `Sequence`.
AWS ParallelCluster no admite incluir tanto un único script como `Sequence` para la misma acción personalizada.

Iam

(Opcional) Especifica una función de instancia o un perfil de instancia que se usará en el nodo principal para anular la función de instancia o el perfil de instancia predeterminados del clúster.

Iam:

```
InstanceRole: string  
InstanceProfile: string  
S3Access:  
- BucketName: string
```

```
EnableWriteAccess: boolean  
KeyName: string  
AdditionalIamPolicies:  
- Policy: string
```

Política de actualización: esta configuración se puede cambiar durante una actualización.

Propiedades de **Iam**

InstanceProfile (Opcional, String)

Especifica un perfil de instancia para anular el perfil de instancia predeterminado del nodo principal. No puede especificar InstanceProfile ni InstanceRole. El formato es `arn:Partition:iam::Account:instance-profile/InstanceProfileName`.

Si se especifica, no se puede especificar la configuración de AdditionalIamPolicies y S3Access.

Le recomendamos que especifique una o ambas opciones de AdditionalIamPolicies configuración, ya que las funciones que se añaden a AWS ParallelCluster menudo requieren nuevos permisos. S3Access

Política de actualización: si se cambia esta configuración, no se permite la actualización.

InstanceRole (Opcional, String)

Especifica un rol de instancia para anular el rol de instancia predeterminado del nodo principal. No puede especificar InstanceProfile ni InstanceRole. El formato es `arn:Partition:iam::Account:role/RoleName`.

Si se especifica, no se puede especificar la configuración de AdditionalIamPolicies y S3Access.

Le recomendamos que especifique una o ambas opciones de AdditionalIamPolicies configuración, ya que las funciones que se añaden AWS ParallelCluster suelen requerir nuevos permisos. S3Access

Política de actualización: esta configuración se puede cambiar durante una actualización.

S3Access

S3Access (Opcional)

Especifica un bucket. Se utiliza para generar políticas que concedan el acceso especificado al bucket.

Si se especifica, no se puede especificar la configuración de InstanceRole y InstanceProfile.

Le recomendamos que especifique una o ambas opciones de AdditionalIamPolicies configuración, ya que las funciones que se añaden AWS ParallelCluster suelen requerir nuevos permisos. S3Access

S3Access:

- BucketName: *string*
- EnableWriteAccess: *boolean*
- KeyName: *string*

Política de actualización: esta configuración se puede cambiar durante una actualización.

BucketName (Requerido, String)

Nombre del bucket de .

Política de actualización: esta configuración se puede cambiar durante una actualización.

KeyName (Opcional, String)

La clave del bucket. El valor predeterminado es "*".

Política de actualización: esta configuración se puede cambiar durante una actualización.

EnableWriteAccess (Opcional, Boolean)

Indica si se ha habilitado el acceso a la escritura en el bucket. El valor predeterminado es false.

Política de actualización: esta configuración se puede cambiar durante una actualización.

AdditionalIamPolicies

AdditionalIamPolicies (Opcional)

Especifica una lista separada de nombres de recursos de Amazon (ARN) de políticas de IAM para Amazon EC2. Esta lista se adjunta a la función raíz utilizada para el nodo principal, además de los permisos requeridos por él AWS ParallelCluster.

El nombre de una política de IAM y su ARN son diferentes. Los nombres no se pueden usar.

Si se especifica, no se puede especificar la configuración de InstanceRole y InstanceProfile.

Se recomienda utilizarlos AdditionalIamPolicies porque AdditionalIamPolicies se añaden a los AWS ParallelCluster permisos necesarios y InstanceRole deben incluir todos los permisos necesarios. Los permisos necesarios a menudo cambian de versión a versión a medida que se añaden características.

No hay valor predeterminado.

AdditionalIamPolicies:

- Policy: *string*

Política de actualización: esta configuración se puede cambiar durante una actualización.

Policy (Opcional, [String])

Lista de políticas de IAM.

Política de actualización: esta configuración se puede cambiar durante una actualización.

Imds

(Opcional) Especifica las propiedades del servicio de metadatos de instancias (IMDS). Para obtener más información, consulte [Cómo funciona la versión 2 del servicio de metadatos de instancias](#) en la Guía del usuario de Amazon EC2.

Imds:

Secured: *boolean*

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Propiedades de **Imds**

Secured (Opcional, Boolean)

Si se establece en `true`, restringe el acceso al IMDS del nodo principal (y a las credenciales del perfil de la instancia) a un subconjunto de superusuarios.

Si se establece en `false`, todos los usuarios del nodo principal tienen acceso al IMDS del nodo principal.

Los siguientes usuarios pueden acceder al IMDS del nodo principal:

- usuario raíz
- usuario administrativo del clúster (`pc-cluster-admin` de forma predeterminada)
- usuario predeterminado específico del sistema operativo (`ec2-user` en Amazon Linux 2 y RedHat, `ubuntu` en Ubuntu 18.04, `centos` en CentOS 7)

El valor predeterminado es `true`.

Los `default` usuarios son responsables de garantizar que un clúster tenga los permisos que necesita para interactuar con AWS los recursos. Si inhabilitas `default` el acceso al IMDS de los usuarios, no AWS ParallelCluster podrás administrar los nodos de procesamiento y dejará de funcionar. No deshabilite el acceso al IMDS del usuario `default`.

Cuando a un usuario se le concede acceso al IMDS del nodo principal, puede usar los permisos incluidos en el [perfil de instancia del nodo principal](#). Por ejemplo, pueden usar estos permisos para lanzar instancias de EC2 o leer la contraseña de un dominio de AD que el clúster está configurado para usar en la autenticación.

Para restringir el acceso al IMDS, AWS ParallelCluster administra una cadena de `iptables`

Los usuarios del clúster con acceso a `sudo` pueden habilitar o deshabilitar de forma selectiva el acceso al IMDS del nodo principal para otros usuarios individuales, incluidos los usuarios `default`, ejecutando el comando:

```
$ sudo /opt/parallelcluster/scripts/imds/imds-access.sh --allow <USERNAME>
```

Puede deshabilitar el acceso al IMDS de los usuarios con la opción `--deny` de este comando.

Si inhabilita el acceso al IMDS del usuario `default` sin saberlo, puede restaurar el permiso mediante la opción `--allow`.

Note

Cualquier personalización de las reglas iptables o ip6tables puede interferir con el mecanismo utilizado para restringir el acceso al IMDS en el nodo principal.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Image

(Opcional) Define una imagen personalizada para el nodo principal.

Image:

CustomAmi: *string*

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Propiedades de Image

CustomAmi (Opcional, String)

Especifica el ID de una AMI personalizada que puede utilizar para el nodo principal en lugar de la AMI publicada predeterminada. Para obtener más información, consulte [AWS ParallelCluster Personalización de AMI](#).

Si la AMI personalizada requiere permisos adicionales para su lanzamiento, estos permisos se deben agregar a las políticas de usuario y de nodo principal.

Por ejemplo, si una AMI personalizada tiene asociada una instantánea cifrada, se requieren las siguientes políticas adicionales tanto en las políticas de usuario como en las del nodo principal:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:DescribeKey",
        "kms:ReEncrypt*",
        "kms:CreateGrant",
        "kms:Decrypt"
      ]
    }
  ]
}
```

```

    ],
    "Resource": [
      "arn:aws:kms:<AWS_REGION>:<AWS_ACCOUNT_ID>:key/<AWS_KMS_KEY_ID>"
    ]
  }
]
}

```

Para solucionar problemas de las advertencias de validación de AMI personalizadas, consulte [Solución de problemas con las AMI de](#) .

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Sección de **Scheduling**

(Obligatorio) Define el programador de trabajos que se usa en el clúster y las instancias de procesamiento que administra el programador de trabajos. Puede utilizar el AWS Batch planificador Slurm o. Cada una admite un conjunto diferente de ajustes y propiedades.

Temas

- [Propiedades de Scheduling](#)
- [AwsBatchQueues](#)
- [SlurmQueues](#)
- [SlurmSettings](#)

Scheduling:

Scheduler: `slurm`

ScalingStrategy: `string`

SlurmSettings:

MungeKeySecretArn: `string`

ScaledownIdleTime: `integer`

QueueUpdateStrategy: `string`

EnableMemoryBasedScheduling: `boolean`

CustomSlurmSettings: `[dict]`

CustomSlurmSettingsIncludeFile: `string`

Database:

Uri: `string`

UserName: `string`

PasswordSecretArn: `string`

DatabaseName: *string*

Dns:

DisableManagedDns: *boolean*

HostedZoneId: *string*

UseEc2Hostnames: *boolean*

SlurmQueues:

- Name: *string*

ComputeSettings:

LocalStorage:

RootVolume:

Size: *integer*

Encrypted: *boolean*

VolumeType: *string*

Iops: *integer*

Throughput: *integer*

EphemeralVolume:

MountDir: *string*

CapacityReservationTarget:

CapacityReservationId: *string*

CapacityReservationResourceGroupArn: *string*

CapacityType: *string*

AllocationStrategy: *string*

JobExclusiveAllocation: *boolean*

CustomSlurmSettings: *dict*

Tags:

- Key: *string*

Value: *string*

HealthChecks:

Gpu:

Enabled: *boolean*

Networking:

SubnetIds:

- *string*

AssignPublicIp: *boolean*

SecurityGroups:

- *string*

AdditionalSecurityGroups:

- *string*

PlacementGroup:

Enabled: *boolean*

Id: *string*

Name: *string*

Proxy:

HttpProxyAddress: *string*

```
ComputeResources:
- Name: string
  InstanceType: string
  Instances:
    - InstanceType: string
  MinCount: integer
  MaxCount: integer
  DynamicNodePriority: integer
  StaticNodePriority: integer
  SpotPrice: float
  DisableSimultaneousMultithreading: boolean
  SchedulableMemory: integer
  HealthChecks:
    Gpu:
      Enabled: boolean
    Efa:
      Enabled: boolean
      GdrSupport: boolean
  CapacityReservationTarget:
    CapacityReservationId: string
    CapacityReservationResourceGroupArn: string
  Networking:
    PlacementGroup:
      Enabled: boolean
      Name: string
    CustomSlurmSettings: dict
  Tags:
    - Key: string
      Value: string
CustomActions:
  OnNodeStart:
    Sequence:
      - Script: string
        Args:
          - string
        Script: string
        Args:
          - string
    OnNodeConfigured:
      Sequence:
        - Script: string
          Args:
            - string
          Script: string
```

Args:

- *string*

Iam:InstanceProfile: *string*InstanceRole: *string*S3Access:

- BucketName: *string*
- EnableWriteAccess: *boolean*
- KeyName: *string*

AdditionalIamPolicies:

- Policy: *string*

Image:CustomAmi: *string*Scheduling:Scheduler: awsbatchAwsBatchQueues:

- Name: *string*
- CapacityType: *string*

Networking:SubnetIds:

- *string*

AssignPublicIp: *boolean*SecurityGroups:

- *string*

AdditionalSecurityGroups:

- *string*

ComputeResources: # this maps to a Batch compute environment (initially we support only 1)

- Name: *string*

InstanceTypes:

- *string*

MinvCpus: *integer*DesiredvCpus: *integer*MaxvCpus: *integer*SpotBidPercentage: *float*

Propiedades de **Scheduling**

Scheduler (Requerido, **String**)

Especifica el tipo de programador que se utiliza. Los valores admitidos son `slurm` y `awsbatch`.

Política de actualización: si se cambia esta configuración, no se permite la actualización.


 Note

awsbatch solo es compatible con el sistema operativo `alinux2` y la plataforma `x86_64`.

ScalingStrategy (Opcional, **String**)

Le permite elegir cómo se escalan los nodos dinámicos de Slurm. Los valores admitidos son `all-or-nothing`, `greedy-all-or-nothing` y `best-effort` el valor por defecto es. `all-or-nothing`

Política de actualización: esta configuración se puede cambiar durante una actualización.

 Note

La estrategia de escalado se aplica solo a los nodos que Slurm va a reanudar, no a los nodos que, finalmente, ya estén en funcionamiento.

- `all-or-nothing` Esta estrategia sigue estrictamente una `all-or-nothing-approach`, destinada a evitar las instancias inactivas al final del proceso de escalado. Funciona sobre una `all-or-nothing` base, lo que significa que se amplía completamente o no se amplía en absoluto. Tenga en cuenta que las instancias lanzadas temporalmente pueden conllevar costes adicionales, ya que los trabajos requieren más de 500 nodos o abarcan varios recursos informáticos. Esta estrategia tiene el rendimiento más bajo de las tres posibles estrategias de escalado. El tiempo de escalado depende de la cantidad de trabajos enviados por cada ejecución del programa de currículum de Slurm. Además, no se puede escalar mucho más allá del límite predeterminado de la cuenta de `RunInstances` recursos por ejecución, que es de 1000 instancias de forma predeterminada. Puede encontrar más información en la documentación sobre la regulación de las [API de AWS EC2](#)
- `greedy-all-or-nothing` Al igual que la `all-or-nothing` estrategia, su objetivo es evitar las instancias inactivas después del escalado. Esta estrategia permite sobreescalar temporalmente durante el proceso de escalado para lograr un rendimiento superior al del `all-or-nothing` enfoque, pero también incluye el mismo límite de escalado de 1000 instancias que el límite de la cuenta de `RunInstances` recursos.

- **best-effort** Esta estrategia prioriza el alto rendimiento, incluso si eso significa que algunas instancias podrían estar inactivas al final del proceso de escalado. Intenta asignar tantos nodos como soliciten los trabajos, pero existe la posibilidad de que no se satisfaga toda la solicitud. A diferencia de las demás estrategias, el enfoque más eficaz consiste en acumular más instancias que el RunInstances límite estándar, a costa de tener recursos inactivos a lo largo de las múltiples ejecuciones de los procesos de escalado.

Cada estrategia está diseñada para satisfacer diferentes necesidades de escalado, lo que le permite seleccionar la que mejor se adapte a sus requisitos y limitaciones específicos.

AwsBatchQueues

(Opcional) La configuración de la AWS Batch cola. Solo se admite una cola. Esto es obligatorio si [Scheduler](#) se establece en `awsbatch`. Para obtener más información sobre el programador de `awsbatch`, consulte la [configuración de red](#) y [AWS Batch \(awsbatch\)](#).

AwsBatchQueues:

```
- Name: string
CapacityType: string
Networking:
  SubnetIds:
    - string
  AssignPublicIp: boolean
  SecurityGroups:
    - string
  AdditionalSecurityGroups:
    - string
ComputeResources: # this maps to a Batch compute environment (initially we support
only 1)
  - Name: string
    InstanceTypes:
      - string
    MinvCpus: integer
    DesiredvCpus: integer
    MaxvCpus: integer
    SpotBidPercentage: float
```

Política de actualización: esta configuración se puede cambiar durante una actualización.

Propiedades de **AwsBatchQueues**

Name (Requerido, **String**)

El nombre de la AWS Batch cola.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

CapacityType (Opcional, **String**)

El tipo de recursos informáticos que utiliza la AWS Batch cola. Los valores admitidos son ONDEMAND, SPOT o. CAPACITY_BLOCK El valor predeterminado es ONDEMAND.

Note

Si establece CapacityType en SPOT, su cuenta debe contener un rol vinculado al servicio de AWSServiceRoleForEC2Spot. Puede crear este rol mediante el siguiente AWS CLI comando.

```
$ aws iam create-service-linked-role --aws-service-name spot.amazonaws.com
```

Para obtener más información, consulte [Rol vinculado al servicio para solicitudes de instancias de spot](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

Networking

(Obligatorio) Define la configuración de red de la AWS Batch cola.

Networking:

SubnetIds:

- *string*

AssignPublicIp: *boolean*

SecurityGroups:

- *string*

AdditionalSecurityGroups:

- *string*

Propiedades de **Networking**

SubnetIds (Requerido, **[String]**)

Especifica el ID de una subred existente para aprovisionar la AWS Batch cola. En la actualidad, solo se admite una subred.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

AssignPublicIp (Opcional, **String**)

Crea o asigna una dirección IP pública a los nodos de la cola. AWS Batch Los valores admitidos son `true` y `false`. El valor predeterminado depende de la subred que haya especificado.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

SecurityGroups (Opcional, **[String]**)

Lista de grupos de seguridad que utiliza la AWS Batch cola. Si no especifica grupos de seguridad, AWS ParallelCluster crea nuevos grupos de seguridad.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

AdditionalSecurityGroups (Opcional, **[String]**)

Lista de grupos de seguridad que utiliza la AWS Batch cola.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

ComputeResources

(Obligatorio) Define la ComputeResources configuración de la AWS Batch cola.

```
ComputeResources: # this maps to a Batch compute environment (initially we support only 1)
- Name: string
  InstanceTypes:
    - string
  MinvCpus: integer
  DesiredvCpus: integer
  MaxvCpus: integer
  SpotBidPercentage: float
```

Propiedades de **ComputeResources**

Name (Requerido, **String**)

El nombre del entorno de cómputo de AWS Batch colas.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

InstanceTypes (Requerido, **[String]**)

El conjunto de tipos de instancias del entorno de AWS Batch cómputo. : el tipo de arquitectura de los tipos de instancia debe ser .

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

MinvCpus (Opcional, **Integer**)

La cantidad mínima de CPU virtuales que puede usar un entorno de AWS Batch procesamiento.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

DesiredVcpus (Opcional, **Integer**)

La cantidad deseada de vCPU en el entorno de procesamiento. AWS Batch AWS Batch ajusta este valor entre `MinvCpus` y en `MaxvCpus` función de la demanda de la cola de trabajos.

[Política de actualización: esta configuración no se analiza durante una actualización.](#)

MaxvCpus (Opcional, **Integer**)

La cantidad máxima de CPU virtuales para el entorno de cómputo. AWS Batch No se puede establecer en un valor inferior a `DesiredVcpus`.

[Política de actualización: esta configuración no se puede reducir durante una actualización.](#)

SpotBidPercentage (Opcional, **Float**)

El porcentaje máximo que puede tener el precio una instancia de spot de EC2 en comparación con el precio bajo demanda para ese tipo de instancia antes de que se lancen las instancias. El valor predeterminado es 100. El rango admitido es de 1 a 100.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

SlurmQueues

(Opcional) Configuración de la cola de Slurm. Esto es obligatorio si [Scheduler](#) se establece en `slurm`.

```
SlurmQueues:
- Name: string
  ComputeSettings:
    LocalStorage:
      RootVolume:
        Size: integer
        Encrypted: boolean
        VolumeType: string
        Iops: integer
        Throughput: integer
      EphemeralVolume:
        MountDir: string
    CapacityReservationTarget:
      CapacityReservationId: string
      CapacityReservationResourceGroupArn: string
    CapacityType: string
    AllocationStrategy: string
    JobExclusiveAllocation: boolean
    CustomSlurmSettings: dict
  Tags:
    - Key: string
      Value: string
  HealthChecks:
    Gpu:
      Enabled: boolean
  Networking:
    SubnetIds:
      - string
    AssignPublicIp: boolean
    SecurityGroups:
      - string
    AdditionalSecurityGroups:
      - string
    PlacementGroup:
      Enabled: boolean
      Id: string
      Name: string
    Proxy:
      HttpProxyAddress: string
```

ComputeResources:

- Name: *string*
 - InstanceType: *string*
 - Instances:
 - InstanceType: *string*
 - MinCount: *integer*
 - MaxCount: *integer*
 - DynamicNodePriority: *integer*
 - StaticNodePriority: *integer*
 - SpotPrice: *float*
 - DisableSimultaneousMultithreading: *boolean*
 - SchedulableMemory: *integer*
 - HealthChecks:
 - Gpu:
 - Enabled: *boolean*
 - Efa:
 - Enabled: *boolean*
 - GdrSupport: *boolean*
 - CapacityReservationTarget:
 - CapacityReservationId: *string*
 - CapacityReservationResourceGroupArn: *string*
 - Networking:
 - PlacementGroup:
 - Enabled: *boolean*
 - Name: *string*
 - CustomSlurmSettings: *dict*
 - Tags:
 - Key: *string*
 - Value: *string*
- CustomActions:
 - OnNodeStart:
 - Sequence:
 - Script: *string*
 - Args:
 - *string*
 - Script: *string*
 - Args:
 - *string*
 - OnNodeConfigured:
 - Sequence:
 - Script: *string*
 - Args:
 - *string*
 - Script: *string*

Args:

- *string*

Iam:

- InstanceProfile:** *string*

- InstanceRole:** *string*

S3Access:

- **BucketName:** *string*

- EnableWriteAccess:** *boolean*

- KeyName:** *string*

AdditionalIamPolicies:

- **Policy:** *string*

Image:

- CustomAmi:** *string*

Política de actualización: para esta configuración de valores de lista, se puede añadir un nuevo valor durante una actualización o se debe detener la flota de computación al eliminar un valor existente.

Propiedades de **SlurmQueues**

Name (Requerido, **String**)

El nombre de la cola Slurm.

Note

El tamaño del clúster puede cambiar durante una actualización. Para obtener más información, consulte [Tamaño de la capacidad del clúster y actualización](#)

Política de actualización: si se cambia esta configuración, no se permite la actualización.

CapacityReservationTarget

Note

CapacityReservationTarget se añade con la AWS ParallelCluster versión 3.3.0.

CapacityReservationTarget:

- CapacityReservationId:** *string*

- CapacityReservationResourceGroupArn:** *string*

Especifica la reserva de capacidad bajo demanda para los recursos informáticos de la cola.

CapacityReservationId (Opcional, **String**)

El ID de la reserva de capacidad existente a la que se destinarán los recursos informáticos de la cola. El identificador puede hacer referencia a un [ODCR](#) o a un [bloque de capacidad para ML](#).

La reserva debe usar la misma plataforma que usa la instancia. Por ejemplo, si sus instancias se ejecutan en `rhel8`, su reserva de capacidad debe ejecutarse en la plataforma Red Hat Enterprise Linux. Para obtener más información, consulte [Plataformas compatibles](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

Note

Si incluye [Instances](#) en la configuración del clúster, debe excluir este ajuste de `CapacityReservationId` de nivel de cola de la configuración.

CapacityReservationResourceGroupArn (Opcional, **String**)

El nombre de recurso de Amazon (ARN) del grupo de recursos que actúa como grupo de reservas de capacidad vinculado a un servicio de reservas de capacidad para los recursos informáticos de la cola. AWS ParallelCluster identifica y utiliza la reserva de capacidad más adecuada del grupo de recursos en función de las siguientes condiciones:

- Si `PlacementGroup` está habilitada en [SlurmQueues/Networking](#) [SlurmQueues/ComputeResources/Networking](#), AWS ParallelCluster selecciona un grupo de recursos que se dirija al tipo de instancia y, si el recurso informático existe, `PlacementGroup` para un recurso informático.


`PlacementGroup` debe dirigirse a uno de los tipos de instancias que se definen en [ComputeResources](#).

- Si `PlacementGroup` no está habilitado en [SlurmQueues/Networking](#) [SlurmQueues/ComputeResources/Networking](#), AWS ParallelCluster selecciona un grupo de recursos que se dirija únicamente al tipo de instancia de un recurso informático, si el recurso informático existe.

El grupo de recursos debe tener al menos un ODCR para cada tipo de instancia reservado en una zona de disponibilidad en todos los recursos informáticos y las zonas de disponibilidad de

la cola. Para obtener más información, consulte [Inicio de instancias con ODCR \(reservas de capacidad bajo demanda\)](#).

Para obtener más información sobre los requisitos de configuración de varias subredes, consulte [Networking/SubnetIds](#).


 Note

En la AWS ParallelCluster versión 3.4.0 se añaden varias zonas de disponibilidad.

Política de actualización: debe haberse detenido la flota de computación o configurado `QueueUpdateStrategy` para poder cambiar esta configuración y proceder a la actualización.

CapacityType (Opcional, **String**)

El tipo de recursos informáticos que utiliza la cola de Slurm. Los valores admitidos son ONDEMAND o SPOT. El valor predeterminado es ONDEMAND.

 Note

Si establece CapacityType en SPOT, su cuenta debe tener un rol vinculado al servicio de AWSServiceRoleForEC2Spot. Puede crear este rol mediante el siguiente AWS CLI comando.

```
$ aws iam create-service-linked-role --aws-service-name spot.amazonaws.com
```

Para obtener más información, consulte [Rol vinculado al servicio para solicitudes de instancias de spot](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

Política de actualización: debe haberse detenido la flota de computación o configurado `QueueUpdateStrategy` para poder cambiar esta configuración y proceder a la actualización.

AllocationStrategy (Opcional, **String**)

Especifique la estrategia de asignación para todos los recursos informáticos definidos en [Instances](#).

Valores válidos: lowest-price | capacity-optimized

Valor predeterminado: `lowest-price`

lowest-price

- Si utiliza `CapacityType = ONDEMAND`, Flota de EC2 utiliza el precio para determinar el orden y lanza primero las de precio más bajo.
- Si utiliza `CapacityType = SPOT`, Flota de EC2 lanza instancias desde el grupo de instancias de spot de menor precio que tenga capacidad disponible. Si un grupo se queda sin capacidad antes de cubrir la capacidad requerida, Flota de EC2 cumplirá su solicitud al lanzar instancias por usted. En concreto, Flota de EC2 lanza instancias desde el grupo de instancias de spot de menor precio que tenga capacidad disponible. Flota EC2 puede lanzar instancias de spot desde varios grupos diferentes.
- Si lo configura `CapacityType = CAPACITY_BLOCK`, no hay estrategias de asignación, por lo que el `AllocationStrategy` parámetro no se puede configurar.

capacity-optimized

- Si ha configurado `CapacityType = ONDEMAND`, `capacity-optimized` no está disponible.
- Si se establece `CapacityType = SPOT`, Flota de EC2 lanza instancias desde grupos de instancias de spot con una capacidad óptima para el número de instancias que se van a lanzar.

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

Note

`AllocationStrategy` es compatible a partir de AWS ParallelCluster versión 3.3.0.

JobExclusiveAllocation (Opcional, **String**)

Si se establece en `true`, el indicador `OverSubscribe` de la partición de Slurm se establece en `EXCLUSIVE`. Cuando `OverSubscribe = EXCLUSIVE`, los trabajos de la partición tienen acceso exclusivo a todos los nodos asignados. Para obtener más información, consulte [EXCLUSIVE](#) en la documentación de Slurm.

Valores válidos: `true` | `false`

Valor predeterminado: `false`

Política de actualización: esta configuración se puede cambiar durante una actualización.

Note

JobExclusiveAllocation es compatible a partir de AWS ParallelCluster versión 3.7.0.

CustomSlurmSettings (Opcional, **Dict**)

Define los ajustes de configuración de la partición de Slurm personalizada (cola).

Especifica un diccionario de pares clave-valor de los parámetros de configuración de Slurm personalizados que se aplican a las colas (particiones).

Cada par clave-valor independiente, por ejemplo Param1: Value1, se agrega por separado al final de la línea de configuración de la partición de Slurm en el formato Param1=Value1.

Solo puede especificar los parámetros de configuración de Slurm que no estén incluidos en la lista de denegaciones de CustomSlurmSettings. Para obtener más información acerca de los parámetros de configuración de Slurm incluidos en la lista de denegaciones, consulte [Parámetros de configuración de Slurm enumerados en denegación para CustomSlurmSettings](#).

AWS ParallelCluster solo comprueba si un parámetro está en una lista de denegaciones. AWS ParallelCluster no valida la sintaxis ni la semántica de los parámetros de Slurm configuración personalizados. Usted es responsable de validar los parámetros de configuración personalizados de Slurm. Los parámetros de configuración personalizados de Slurm no válidos pueden provocar errores en los daemons de Slurm y provocar errores en la creación y actualización del clúster.

Para obtener más información sobre cómo especificar parámetros de Slurm configuración personalizados con AWS ParallelCluster, consulte [Personalización de la configuración de Slurm](#).

Para obtener más información sobre los parámetros de configuración de Slurm, consulte [slurm.conf](#) en la documentación de Slurm.

Política de actualización: esta configuración se puede cambiar durante una actualización.

Note

CustomSlurmSettings es compatible a partir de AWS ParallelCluster versión 3.6.0.

(Cadena) Opcional.

Una lista de pares clave-valor de etiqueta. Las etiquetas de [ComputeResource](#) anulan las etiquetas duplicadas especificadas en [Sección de Tags](#) o en SlurmQueues/Tags.

Key (Opcional, **String**)

La clave de la etiqueta.

Value (Opcional, **String**)

El valor de la etiqueta.


[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

HealthChecks (Opcional)

Especifique las comprobaciones de estado de los nodos de cómputo en todos los recursos de cómputo de la cola.

Gpu (Opcional)

Especifica las comprobaciones de estado de la GPU en todos los recursos informáticos de una cola.

 Note

AWS ParallelCluster no admite HealthChecks/Gpu en los nodos que utilizan sistemas operativos a linux2 ARM. Estas plataformas no son compatibles con el [administrador de GPU para centros de datos de NVIDIA \(DCGM\)](#).

Enabled (Opcional, **Boolean**)

Si AWS ParallelCluster realiza comprobaciones de estado de la GPU en los nodos de cómputo. El valor predeterminado es false.

Comportamiento de la comprobación de estado de la **Gpu**

- Si Gpu/Enabled está establecido en true, AWS ParallelCluster realiza comprobaciones de estado de la GPU en los recursos de cómputo de la cola.

- La comprobación de estado de la Gpu realiza comprobaciones del estado de la GPU en los recursos informáticos para evitar que se envíen trabajos en nodos con una GPU degradada.
- Si un nodo de cómputo no pasa una comprobación de estado de la Gpu, el estado del nodo de computación cambia a DRAIN. Los nuevos trabajos no comienzan en este nodo. Los trabajos existentes se ejecutan hasta su finalización. Una vez finalizados todos los trabajos en ejecución, el nodo de cómputo finaliza si es un nodo dinámico y se reemplaza si es un nodo estático.
- La duración de la comprobación de estado de la Gpu depende del tipo de instancia seleccionada, la cantidad de GPU de la instancia y la cantidad de objetivos de la comprobación de estado de la Gpu (equivalente a la cantidad de objetivos de GPU de trabajo). En una instancia con 8 GPU, la duración normal es inferior a 3 minutos.
- Si la comprobación de estado de la Gpu se ejecuta en una instancia que no es compatible, se cierra y el trabajo se ejecuta en el nodo de cómputo. Por ejemplo, si una instancia no tiene una GPU o si tiene una GPU pero no es una GPU de NVIDIA, la comprobación de estado finaliza y el trabajo se ejecuta en el nodo de cómputo. Solo se admiten las GPU NVIDIA.
- La comprobación de estado de la Gpu utiliza la herramienta `dcgmi` para realizar comprobaciones de estado en un nodo y sigue los siguientes pasos:

Cuando la comprobación de estado de la Gpu comienza en un nodo:

1. Detecta si los servicios `nvidia-dcgm` y `nvidia-fabricmanager` se están ejecutando.
2. Si estos servicios no se están ejecutando, la comprobación de estado de la Gpu los inicia.
3. Detecta si el modo de persistencia está activado.
4. Si el modo de persistencia no está activado, la comprobación de estado de la Gpu lo habilita.


Al final de la comprobación de estado, la comprobación de estado de la Gpu restablece estos servicios y recursos a su estado inicial.

- Si el trabajo está asignado a un conjunto específico de GPU de nodos, la comprobación de estado de la Gpu solo se ejecuta en ese conjunto específico. De lo contrario, la comprobación de estado de la Gpu se ejecuta en todas las GPU del nodo.
- Si un nodo de cómputo recibe 2 o más solicitudes de comprobación de estado de la Gpu al mismo tiempo, solo se ejecuta la primera comprobación de estado y se omiten las demás. Este también es el caso de las comprobaciones de estado dirigidas a las GPU de los nodos.

Puede consultar los archivos de registro para obtener información adicional sobre esta situación.

- El registro de comprobación de estado de un nodo informático específico está disponible en el archivo `/var/log/parallelcluster/slurm_health_check.log`. El archivo está disponible en Amazon CloudWatch, en el grupo de CloudWatch registros del clúster, donde puede encontrar:
 - Detalles sobre la acción que lleva a cabo la comprobación de estado de la Gpu, incluida la activación y desactivación de los servicios y el modo de persistencia.
 - El identificador de la GPU, el ID de serie y el UUID.
 - El resultado del comprobación de estado.

Política de actualización: esta configuración se puede cambiar durante una actualización.

 Note

HealthChecksse admite a partir de la AWS ParallelCluster versión 3.6.0.

Networking

(Obligatorio) Define la configuración de red de la cola de Slurm.

```
Networking:
  SubnetIds:
    - string
  AssignPublicIp: boolean
  SecurityGroups:
    - string
  AdditionalSecurityGroups:
    - string
  PlacementGroup:
    Enabled: boolean
    Id: string
    Name: string
  Proxy:
    HttpProxyAddress: string
```

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Propiedades de **Networking**

SubnetIds (Requerido, **[String]**)

Los ID de las subredes existentes en las que aprovisiona la cola de Slurm.

Si configura los tipos de instancias en [SlurmQueues/ComputeResources/InstanceType](#), solo puede definir una subred.

Si configura los tipos de instancias en [SlurmQueues/ComputeResources/Instances](#), puede definir una o varias subredes.

Si usa varias subredes, todas las subredes definidas para una cola deben estar en la misma VPC y cada subred en una zona de disponibilidad (AZ) independiente.

Por ejemplo, supongamos que define la subred-1 y la subred-2 para la cola.

subnet-1 y no subnet-2 pueden estar ambos en AZ-1.

subnet-1 puede estar en AZ-1 y subnet-2 puede estar en AZ-2.

Si configura solo un tipo de instancia y desea usar varias subredes, defina el tipo de instancia en Instances lugar de InstanceType.

Por ejemplo, defina ComputeResources/Instances/InstanceType = `instance.type` en lugar de ComputeResources/InstanceType = `instance.type`.

Note

Elastic Fabric Adapter (EFA) no es compatible con diferentes zonas de disponibilidad.

El uso de varias zonas de disponibilidad puede provocar un aumento de la latencia de las redes de almacenamiento y aumentar los costes de transferencia de datos entre zonas de disponibilidad. Por ejemplo, esto podría ocurrir cuando una instancia accede al almacenamiento de archivos ubicado en una zona de disponibilidad diferente. Para obtener más información, consulte [Transferencia de datos dentro de la misma Región de AWS](#).

Actualizaciones del clúster para pasar del uso de una sola subred a varias subredes:

- Supongamos que la definición de subred de un clúster se define con una única subred y un sistema de archivos FSx for Lustre AWS ParallelCluster administrado. Por lo tanto, no puede

actualizar este clúster directamente con una definición de ID de subred actualizada. Para actualizar el clúster, primero debe cambiar el sistema de archivos administrado por un sistema de archivos externo. Para obtener más información, consulte [Convierta el almacenamiento AWS ParallelCluster gestionado en almacenamiento externo](#).

- Supongamos que la definición de subred de un clúster se define con una única subred y un sistema de archivos Amazon EFS externo si no existen objetivos de montaje de EFS para todas las AZ de las múltiples subredes definidas que se van a añadir. Por lo tanto, no puede actualizar este clúster directamente con una definición de ID de subred actualizada. Para actualizar el clúster o crear uno, primero debe crear todos los destinos de montaje de todas las AZ de las múltiples subredes definidas.

[Las zonas de disponibilidad y las reservas de capacidad del clúster se definen en Arn: CapacityReservation ResourceGroup](#)

- No puede crear un clúster si no hay superposición entre el conjunto de tipos de instancias y zonas de disponibilidad que cubre el grupo de recursos de reserva de capacidad definido y el conjunto de tipos de instancias y zonas de disponibilidad definidos para la cola.
- Puede crear un clúster si hay una superposición parcial entre el conjunto de tipos de instancias y zonas de disponibilidad que cubre el grupo de recursos de reserva de capacidad definido y el conjunto de tipos de instancias y zonas de disponibilidad definidos para la cola. AWS ParallelCluster envía un mensaje de advertencia sobre la superposición parcial en este caso.
- Para obtener más información, consulte [Inicio de instancias con ODCR \(reservas de capacidad bajo demanda\)](#).

Note

En la AWS ParallelCluster versión 3.4.0 se añaden varias zonas de disponibilidad.

Warning

Esta advertencia se aplica a todas las AWS ParallelCluster versiones 3.x.y anteriores a la 3.3.1. AWS ParallelCluster la versión 3.3.1 no se ve afectada si se cambia este parámetro.

Para AWS ParallelCluster 3 versiones anteriores a la 3.3.1:

Al cambiar este parámetro y actualizar un clúster, se crea un nuevo sistema de archivos FSx para Lustre administrado y se elimina el sistema de archivos FSx para Lustre

administrado existente sin conservar los datos existentes. Esto provoca la pérdida de datos. Antes de continuar, asegúrese de hacer una copia de seguridad de los datos del sistema de archivos FSx para Lustre existente si desea conservar los datos. Para obtener más información, consulte [Working with backups](#) en la Guía del usuario de Amazon FSx para Lustre.

Si se agrega un nuevo valor de subred, [Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

Si se elimina un valor de subred, [Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

AssignPublicIp (Opcional, **String**)

Crea o asigna una dirección IP pública a los nodos de la cola de Slurm. Los valores admitidos son `true` y `false`. La subred que especifique determina el valor predeterminado. Una subred con direcciones IP públicas asigna direcciones IP públicas de forma predeterminada.

Si define un tipo de hpc6id instancia p4d OR u otro tipo de instancia que tenga varias interfaces de red o una tarjeta de interfaz de red, debe configurar [HeadNode/Networking/ElasticIp](#) `true` para proporcionar acceso público. AWS Las IP públicas solo se pueden asignar a instancias lanzadas con una única interfaz de red. En este caso, le recomendamos que utilice una [puerta de enlace NAT](#) para proporcionar acceso público a los nodos de computación del clúster. En este caso, establezca `AssignPublicIp` en `false`. Para obtener más información sobre las direcciones IP, consulte [Asignación de una dirección IPv4 pública durante el lanzamiento de una instancia](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

SecurityGroups (Opcional, **[String]**)

Una lista de grupos de seguridad que se utilizarán para el entorno. Si no se especifica ningún grupo de seguridad, AWS ParallelCluster crea grupos de seguridad automáticamente.

Compruebe que los grupos de seguridad estén configurados correctamente para sus [SharedStorage](#) sistemas.

⚠ Warning

Esta advertencia se aplica a los tres. *x*. *y* AWS ParallelCluster versiones anteriores a la versión 3.3.0. AWS ParallelCluster la versión 3.3.0 no se ve afectada si se cambia este parámetro.

Para AWS ParallelCluster 3 versiones anteriores a la 3.3.0:

Al cambiar este parámetro y actualizar un clúster, se crea un nuevo sistema de archivos FSx para Lustre administrado y se elimina el sistema de archivos FSx para Lustre administrado existente sin conservar los datos existentes. Esto provoca la pérdida de datos. Asegúrese de hacer una copia de seguridad de los datos del sistema de archivos FSx para Lustre existente si desea conservar los datos. Para obtener más información, consulte [Working with backups](#) en la Guía del usuario de Amazon FSx para Lustre.

⚠ Warning

Si habilita [EFA](#) para las instancias de procesamiento, asegúrese de que las instancias habilitadas para EFA formen parte de un grupo de seguridad que permita todo el tráfico entrante y saliente hacia sí mismo.

Política de actualización: esta configuración se puede cambiar durante una actualización.

AdditionalSecurityGroups (Opcional, **[String]**)

Una lista de grupos de seguridad adicionales que se utilizarán para la cola de Slurm.

Política de actualización: esta configuración se puede cambiar durante una actualización.

PlacementGroup (Opcional)

Especifica la configuración del grupo de ubicación de la cola de Slurm.

PlacementGroup:

Enabled: *boolean*

Id: *string*

Name: *string*

Política de actualización: todos los nodos de procesamiento deben estar detenidos para poder eliminar un grupo de ubicación administrado. Debe haberse detenido la flota de computación o

[configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

Enabled (Opcional, **Boolean**)

Indica si se utiliza un grupo de ubicaciones para la cola de Slurm. El valor predeterminado es `false`.

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

Id (Opcional, **String**)

El nombre del grupo de ubicación de un grupo con ubicación en clúster existente que utiliza la cola de Slurm. Asegúrese de proporcionar el nombre del grupo de ubicación y no el ID.

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

Name (Opcional, **String**)

El nombre del grupo de ubicación de un grupo con ubicación en clúster existente que utiliza la cola de Slurm. Asegúrese de proporcionar el nombre del grupo de ubicación y no el ID.

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

Note

- Si `PlacementGroup/Enabled` se establece en `true`, sin un `Name` o `Id` definidos, a cada recurso informático se le asigna su propio grupo de ubicación administrado, a menos que se defina [ComputeResources/Networking/PlacementGroup](#) para anular esta configuración.
- A partir de AWS ParallelCluster la versión 3.3.0, [Name](#) se agregó [SlurmQueuesNetworkingPlacementGroup](#) como alternativa preferida a [SlurmQueues//Networking/PlacementGroup](#). [Id](#)

[PlacementGroup/Id](#) y [PlacementGroup/Name](#) son equivalentes. Puede utilizar cualquiera de los dos.

Si incluye tanto [PlacementGroup](#) como [PlacementGroup/Name](#), [Id](#) se produce un AWS ParallelCluster error. Debe elegir una u otra.

No necesita actualizar el clúster para usar [PlacementGroup/Name](#).

Proxy (Opcional)

Especifica la configuración del proxy de la cola de Slurm.

Proxy:

HttpProxyAddress: *string*

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

HttpProxyAddress (Opcional, **String**)

Define un servidor proxy HTTP o HTTPS para la cola de Slurm. Por lo general, es `https://x.x.x.x:8080`.

No hay valor predeterminado.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Image

(Opcional) Especifica la imagen que se utilizará en la cola de Slurm. Para usar la misma AMI para todos los nodos, utilice la [CustomAmi](#) configuración de la [Image](#) sección.

Image:

CustomAmi: *string*

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Propiedades de **Image**

CustomAmi (Opcional, **String**)

La AMI que se utilizará en la cola de Slurm en lugar de las AMI predeterminadas. Puede usar el comando de CLI `pcluster` para ver una lista de las AMI predeterminadas.

Note

La AMI debe estar basada en el mismo sistema operativo que utiliza el nodo principal.

pcluster list-official-images

Si la AMI personalizada requiere permisos adicionales para su lanzamiento, debe agregar estos permisos a la política del nodo principal.

Por ejemplo, si una AMI personalizada tiene asociada una instantánea cifrada, se requieren las siguientes políticas adicionales en las políticas del nodo principal.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:DescribeKey",
        "kms:ReEncrypt*",
        "kms:CreateGrant",
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:<AWS_REGION>:<AWS_ACCOUNT_ID>:key/<AWS_KMS_KEY_ID>"
      ]
    }
  ]
}
```

Para solucionar problemas de las advertencias de validación de AMI personalizadas, consulte [Solución de problemas con las AMI de](#) .

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

ComputeResources

(Obligatorio) Define la configuración de ComputeResources de la cola de Slurm.

Note

El tamaño del clúster puede cambiar durante una actualización. Para obtener más información, consulte [Tamaño de la capacidad del clúster y actualización](#)

ComputeResources:

- **Name:** *string*
- InstanceType:** *string*
- Instances:**
 - **InstanceType:** *string*
- MinCount:** *integer*
- MaxCount:** *integer*
- DynamicNodePriority:** *integer*
- StaticNodePriority:** *integer*
- SpotPrice:** *float*
- DisableSimultaneousMultithreading:** *boolean*
- SchedulableMemory:** *integer*
- HealthChecks:**
 - Gpu:**
 - Enabled:** *boolean*
- Efa:**
 - Enabled:** *boolean*
 - GdrSupport:** *boolean*
- CapacityReservationTarget:**
 - CapacityReservationId:** *string*
 - CapacityReservationResourceGroupArn:** *string*
- Networking:**
 - PlacementGroup:**
 - Enabled:** *boolean*
 - Name:** *string*
- CustomSlurmSettings:** *dict*
- Tags:**
 - **Key:** *string*
 - Value:** *string*

Política de actualización: para esta configuración de valores de lista, se puede añadir un nuevo valor durante una actualización o se debe detener la flota de computación al eliminar un valor existente.

Propiedades de **ComputeResources**

Name (Requerido, **String**)

El nombre del entorno informático. El nombre puede tener un máximo de 25 caracteres.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

InstanceType (Requerido, **String**)

El tipo de instancia que se usa en este recurso informático de Slurm. Todos los tipos de instancias de un clúster deben usar la misma arquitectura de procesador. Las instancias pueden usar la arquitectura `x86_64` o `arm64`.

La configuración del clúster debe definir una [InstanceType](#) o varias [instancias](#). Si ambas están definidas, se produce un AWS ParallelCluster error.

Cuando se define `InstanceType`, no se pueden definir varias subredes. Si configura solo un tipo de instancia y desea usar varias subredes, defina el tipo de instancia en `Instances` lugar de en `InstanceType`. Para obtener más información, consulte [Networking](#) / [SubnetIds](#).

Si define un tipo de `hpc6id` instancia `p4d` o, u otro tipo de instancia que tenga varias interfaces de red o una tarjeta de interfaz de red, debe lanzar las instancias de procesamiento en una subred privada, tal y como se describe en [AWS ParallelCluster con dos subredes](#). AWS Las IP públicas solo se pueden asignar a las instancias que se lanzan con una única interfaz de red. Para obtener más información, consulte [Asignación de una dirección IPv4 pública durante el lanzamiento de una instancia](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

Instances (Obligatorio)

Especifica la lista de tipos de instancias de un recurso informático. Para especificar la estrategia de asignación de la lista de tipos de instancias, consulte [AllocationStrategy](#).

La configuración del clúster debe definir un [InstanceType](#) o [Instances](#). Si ambas opciones están definidas, AWS ParallelCluster produce un error.

Para obtener más información, consulte [Asignación de varios tipos de instancias con Slurm](#).

```
Instances:  
- InstanceType: string
```


Note

A partir de AWS ParallelCluster la versión 3.7.0, se `EnableMemoryBasedScheduling` puede habilitar si configuras varios tipos de [instancias en Instances](#).

Para AWS ParallelCluster las versiones 3.2.0 a 3.6. **x**, no se `EnableMemoryBasedScheduling` puede habilitar si configuras varios tipos de [instancias en Instances](#).

Política de actualización: para esta configuración de valores de lista, se puede añadir un nuevo valor durante una actualización o se debe detener la flota de computación al eliminar un valor existente.

InstanceType (Requerido, **String**)

El tipo de instancia que se va a usar en este recurso informático de Slurm. Todos los tipos de instancias de un clúster deben usar la misma arquitectura de procesador, ya sea x86_64 o arm64.

Los tipos de instancias que se enumeran en [Instances](#) deben tener:

- El mismo número de vCPU o, si [DisableSimultaneousMultithreading](#) se ha establecido en `true`, el mismo número de núcleos.
- El mismo número de aceleradores de los mismos fabricantes.
- Compatible con EFA, si [Efa/Enabled](#) se establece en `true`.

Los tipos de instancias que aparecen en la lista [Instances](#) pueden tener:

- Cantidad de memoria diferente.

En este caso, la memoria mínima debe configurarse como recurso de Slurm consumible.

Note

A partir de AWS ParallelCluster la versión 3.7.0, se `EnableMemoryBasedScheduling` puede habilitar si configuras varios tipos de [instancias en Instances](#).

Para AWS ParallelCluster las versiones 3.2.0 a 3.6. **x**, no se `EnableMemoryBasedScheduling` puede habilitar si configuras varios tipos de [instancias en Instances](#).

- Tarjetas de red diferentes.


En este caso, el número de interfaces de red configuradas para el recurso informático viene definido por el tipo de instancia con el menor número de tarjetas de red.

- Ancho de banda de la red diferente.
- Tamaño de almacén de instancias diferente.

Si define un tipo de hpc6id instancia p4d o, u otro tipo de instancia que tenga varias interfaces de red o una tarjeta de interfaz de red, debe lanzar las instancias de procesamiento en una subred privada, tal y como se describe en [AWS ParallelCluster con dos subredes](#). AWS Las IP públicas solo se pueden asignar a instancias lanzadas con una única interfaz de red.

Para obtener más información, consulte [Asignación de una dirección IPv4 pública durante el lanzamiento de una instancia](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.


[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

 Note

Instancesse admite a partir de la AWS ParallelCluster versión 3.3.0.

MinCount (Opcional, **Integer**)

El número mínimo de instancias que utiliza el recurso informático de Slurm. El valor predeterminado es 0.

 Note

El tamaño del clúster puede cambiar durante una actualización. Para obtener más información, consulte [Tamaño de la capacidad del clúster y actualización](#)


[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

MaxCount (Opcional, **Integer**)

El número máximo de instancias que utiliza el recurso informático de Slurm. El valor predeterminado es 10.

Cuando se usa `CapacityType = CAPACITY_BLOCK`, `MaxCount` debe ser igual o superior a `MinCount 0`, ya que todas las instancias que forman parte de la reserva del bloque de capacidad se administran como nodos estáticos.

En el momento de la creación del clúster, el nodo principal espera a que todos los nodos estáticos estén listos para indicar que la creación del clúster se ha realizado correctamente. Sin embargo, al utilizarlos `CapacityType = CAPACITY_BLOCK`, los nodos que forman parte de los recursos informáticos asociados a los bloques de capacidad no se tendrán en cuenta para esta comprobación. El clúster se creará aunque no estén activos todos los bloques de capacidad configurados.


 Note

El tamaño del clúster puede cambiar durante una actualización. Para obtener más información, consulte [Tamaño de la capacidad del clúster y actualización](#)

DynamicNodePriority (Opcional, **Integer**)

La prioridad de los nodos dinámicos en un recurso informático de cola. La prioridad se asigna al parámetro de configuración [Weight](#) del nodo de Slurm para los nodos dinámicos del recurso informático. El valor predeterminado es `1000`.

Slurm prioriza primero los nodos con los valores de `Weight` más bajos.

 Warning

El uso de muchos valores de `Weight` diferentes en una partición (cola) de Slurm puede reducir el ritmo de programación de trabajos en la cola.

En AWS ParallelCluster las versiones anteriores a la 3.7.0, a los nodos estáticos y dinámicos se les asignaba el mismo peso predeterminado de `1`. En este caso, Slurm podría priorizar los nodos dinámicos inactivos sobre los nodos estáticos inactivos debido al esquema de nomenclatura de los nodos estáticos y dinámicos. Cuando todo lo demás es igual, Slurm programa los nodos alfabéticamente por nombre.

Note

DynamicNodePriority se agregó en la AWS ParallelCluster versión 3.7.0.

Política de actualización: esta configuración se puede cambiar durante una actualización.

StaticNodePriority (Opcional, **Integer**)

La prioridad de los nodos estáticos en un recurso de cómputo de cola. La prioridad se asigna al parámetro de configuración [Weight](#) del nodo de Slurm para los nodos estáticos del recurso informático. El valor predeterminado es 1.

Slurm prioriza primero los nodos con los valores de Weight más bajos.

Warning

El uso de muchos valores de Weight diferentes en una partición (cola) de Slurm puede reducir el ritmo de programación de trabajos en la cola.

Note

StaticNodePriority se añadió en la AWS ParallelCluster versión 3.7.0.

Política de actualización: esta configuración se puede cambiar durante una actualización.

SpotPrice (Opcional, **Float**)

El precio máximo que se paga por una instancia de spot de EC2 antes del lanzamiento de cualquier instancia. El valor predeterminado es precio bajo demanda.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

DisableSimultaneousMultithreading (Opcional, **Boolean**)

Si true, el subprocesamiento múltiple en los nodos de la cola de Slurm está deshabilitado. El valor predeterminado es false.

No todos los tipos de instancias pueden deshabilitar los subprocesos múltiples. Para obtener una lista de los tipos de instancias que admiten la desactivación de subprocesos múltiples, consulte [Núcleos de CPU y subprocesos para cada núcleo de CPU por tipo de instancia](#) en la Guía del usuario de Amazon EC2.

Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.

SchedulableMemory (Opcional, **Integer**)

La cantidad de memoria en MiB que se configura en el parámetro de Slurm `RealMemory` para los nodos de computación de un recurso informático. Este valor es el límite superior de la memoria de nodo disponible para los trabajos cuando se habilita [SlurmSettings/EnableMemoryBasedScheduling](#). El valor predeterminado es el 95 por ciento de la memoria que aparece en los tipos de instancias de Amazon EC2 y que devuelven los tipos de API de Amazon DescribeInstance EC2. Asegúrese de convertir los valores expresados en GiB a MiB.

Valores admitidos: 1-EC2Memory

`EC2Memory` es la memoria (en MiB) que aparece en los tipos de instancias de Amazon EC2 y que devuelven los tipos de API de Amazon EC2. DescribeInstance Asegúrese de convertir los valores expresados en GiB a MiB.

Esta opción es más relevante cuando se habilita [SlurmSettings/EnableMemoryBasedScheduling](#). Para obtener más información, consulte [Slurm programación basada en memoria](#).

Note

`SchedulableMemory` es compatible a partir de AWS ParallelCluster versión 3.2.0. A partir de la versión 3.2.0, de forma predeterminada, se AWS ParallelCluster configura `RealMemory` para los nodos de Slurm cómputo el 95 por ciento de la memoria que devuelve la API de Amazon EC2. `DescribeInstanceTypes` Esta configuración es independiente del valor de `EnableMemoryBasedScheduling`.

Política de actualización: debe haberse detenido la flota de computación o configurado `QueueUpdateStrategy` para poder cambiar esta configuración y proceder a la actualización.

HealthChecks (opcional)

Especifique las comprobaciones de estado de un recurso informático.

Gpu (opcional)

Especifique las comprobaciones de estado de la GPU en un recurso informático.

Enabled (Opcional, Boolean)

Si AWS ParallelCluster realiza comprobaciones de estado de la GPU al calcular un recurso en una cola. El valor predeterminado es `false`.

Note

AWS ParallelCluster no admite `HealthChecks/Gpu` en los nodos que utilizan sistemas operativos `alinux2 ARM`. Estas plataformas no son compatibles con el [administrador de GPU para centros de datos de NVIDIA \(DCGM\)](#).

Comportamiento de la comprobación de estado de la Gpu

- Si `Gpu/Enabled` está establecido en `true`, AWS ParallelCluster realiza comprobaciones de estado de la GPU en un recurso informático.
- La comprobación de estado de la `Gpu` realiza comprobaciones de estado de un recurso informático para evitar que se envíen trabajos en nodos con una GPU degradada.
- Si un nodo de cómputo no pasa una comprobación de estado de la `Gpu`, el estado del nodo de computación cambia a `DRAIN`. Los nuevos trabajos no comienzan en este nodo. Los trabajos existentes se ejecutan hasta su finalización. Una vez finalizados todos los trabajos en ejecución, el nodo de cómputo finaliza si es un nodo dinámico y se reemplaza si es un nodo estático.
- La duración de la comprobación de estado de la `Gpu` depende del tipo de instancia seleccionada, la cantidad de GPU de la instancia y la cantidad de objetivos de la comprobación de estado de la `Gpu` (equivalente a la cantidad de objetivos de GPU de trabajo). En una instancia con 8 GPU, la duración normal es inferior a 3 minutos.
- Si la comprobación de estado de la `Gpu` se ejecuta en una instancia que no es compatible, se cierra y el trabajo se ejecuta en el nodo de cómputo. Por ejemplo, si una instancia no tiene una GPU o si tiene una GPU pero no es una GPU de NVIDIA, la comprobación de estado finaliza y el trabajo se ejecuta en el nodo de cómputo. Solo se admiten las GPU NVIDIA.

- La comprobación de estado de la Gpu utiliza la herramienta `dcgmi` para realizar comprobaciones de estado en un nodo y sigue los siguientes pasos:


Cuando la comprobación de estado de la Gpu comienza en un nodo:

1. Detecta si los servicios `nvidia-dcgm` y `nvidia-fabricmanager` se están ejecutando.
2. Si estos servicios no se están ejecutando, la comprobación de estado de la Gpu los inicia.
3. Detecta si el modo de persistencia está activado.
4. Si el modo de persistencia no está activado, la comprobación de estado de la Gpu lo habilita.

Al final de la comprobación de estado, la comprobación de estado de la Gpu restablece estos servicios y recursos a su estado inicial.

- Si el trabajo está asignado a un conjunto específico de GPU de nodos, la comprobación de estado de la Gpu solo se ejecuta en ese conjunto específico. De lo contrario, la comprobación de estado de la Gpu se ejecuta en todas las GPU del nodo.
- Si un nodo de cómputo recibe 2 o más solicitudes de comprobación de estado de la Gpu al mismo tiempo, solo se ejecuta la primera comprobación de estado y se omiten las demás. Este también es el caso de las comprobaciones de estado dirigidas a las GPU de los nodos. Puede consultar los archivos de registro para obtener información adicional sobre esta situación.
- El registro de comprobación de estado de un nodo informático específico está disponible en el archivo `/var/log/parallelcluster/slurm_health_check.log`. Este archivo está disponible en Amazon CloudWatch, en el grupo de CloudWatch registros del clúster, donde puede encontrar:
 - Detalles sobre la acción que lleva a cabo la comprobación de estado de la Gpu, incluida la activación y desactivación de los servicios y el modo de persistencia.
 - El identificador de la GPU, el ID de serie y el UUID.
 - El resultado de la comprobación de estado.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

 Note

HealthChecksse admite a partir de la AWS ParallelCluster versión 3.6.0.

Efa (opcional)

Especifica la configuración Elastic Fabric Adapter (EFA) para los nodos de la cola de Slurm.

Efa:

Enabled: *boolean*

GdrSupport: *boolean*

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Enabled (Opcional, Boolean)

Especifica que Elastic Fabric Adapter (EFA) esté habilitado. Para ver la lista de instancias de EC2 compatibles con EFA, consulte [Tipos de instancias admitidas](#) en la Guía del usuario para instancias de Linux de Amazon EC2. Para obtener más información, consulte [Elastic Fabric Adapter](#). Le recomendamos que utilice un clúster [SlurmQueues/Networking/PlacementGroup](#) para minimizar las latencias entre instancias.

El valor predeterminado es `false`.

Note

Elastic Fabric Adapter (EFA) no es compatible con diferentes zonas de disponibilidad. Para obtener más información, consulte [SubnetIds](#).

Warning


Si va a definir un grupo de seguridad personalizado en [SecurityGroups](#), asegúrese de que las instancias habilitadas para EFA sean miembros de un grupo de seguridad que permita que todo el tráfico entrante y saliente se dirija a sí mismo.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

GdrSupport (Opcional, Boolean)

(Opcional) A partir de AWS ParallelCluster versión 2.11.3, esta configuración no tiene efecto. La compatibilidad con el Elastic Fabric Adapter (EFA) para GPUDirect RDMA (acceso directo

a memoria) está habilitada para los nodos de procesamiento y siempre está habilitada si el tipo de instancia lo admite.

 Note

AWS ParallelCluster versiones 3.0.0 a 3.0.1: Support for GPUDirect RDMA está habilitado para los recursos de cómputo. Slurm La compatibilidad de GPUDirect RDMA la admiten tipos de instancias concretos (p4d.24xlarge) en sistemas operativos específicos ([Os](#) es alinux2, centos7, ubuntu1804 o ubuntu2004). El valor predeterminado es false.

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

CapacityReservationTarget

[CapacityReservationTarget](#):

[CapacityReservationId](#): *string*

[CapacityReservationResourceGroupArn](#): *string*

Especifica la reserva de capacidad bajo demanda que se utilizará para el recurso informático.

CapacityReservationId (Opcional, **String**)

El ID de la reserva de capacidad existente a la que se destinarán los recursos informáticos de la cola. [El identificador puede hacer referencia a un ODCR o a un bloque de capacidad para ML.](#)

Si este parámetro se especifica a nivel de recurso informático, si InstanceType es opcional, se recuperará automáticamente de la reserva.

CapacityReservationResourceGroupArn (Opcional, **String**)

Indica el nombre de recurso de Amazon (ARN) del grupo de recursos que actúa como grupo de reservas de capacidad vinculado a un servicio de reservas de capacidad para el recurso de cómputo. AWS ParallelCluster identifica y utiliza la reserva de capacidad más adecuada del grupo. El grupo de recursos debe tener al menos un ODCR para cada tipo de instancia que aparezca en la lista del recurso informático. Para obtener más información, consulte [Inicio de instancias con ODCR \(reservas de capacidad bajo demanda\)](#).

- Si PlacementGroup está habilitado en [SlurmQueues/Networking/SlurmQueuesComputeResources/Networking](#), AWS ParallelCluster selecciona un grupo de recursos que se dirija al tipo de instancia yPlacementGroup, si existe, a un recurso informático.

PlacementGroup debe dirigirse a uno de los tipos de instancias definidos en [ComputeResources](#).

- Si PlacementGroup no está habilitado en [SlurmQueues/Networking/SlurmQueues/ComputeResources/Networking](#), AWS ParallelCluster selecciona un grupo de recursos que se dirija solo al tipo de instancia de un recurso informático, si existe.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Note

CapacityReservationTarget se añade con AWS ParallelCluster versión 3.3.0.

Networking

[Networking](#):

[PlacementGroup](#):

[Enabled](#): *boolean*

[Name](#): *string*

Política de actualización: todos los nodos de procesamiento deben estar detenidos para poder eliminar un grupo de ubicación administrado. Debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

PlacementGroup (opcional)

Especifica la configuración del grupo de ubicación para el recurso de cómputo.

Enabled (Opcional, **Boolean**)

Indica si se utiliza un grupo de ubicación para el recurso informático.

- Si se establece en true, sin un Name definido, a ese recurso informático se le asigna su propio grupo de ubicación administrado, independientemente de la configuración de [SlurmQueues/Networking/PlacementGroup](#).

- Si se establece en `true`, con un `Name` definido, a ese recurso informático se le asigna el grupo de ubicación indicado, independientemente de la configuración de `SlurmQueues/Networking/PlacementGroup`.

Política de actualización: debe haberse detenido la flota de computación o configurado `QueueUpdateStrategy` para poder cambiar esta configuración y proceder a la actualización.

Name (Opcional, **String**)

El nombre del grupo de ubicación de un grupo con ubicación en clúster existente que se usa para el recurso informático.

Política de actualización: debe haberse detenido la flota de computación o configurado `QueueUpdateStrategy` para poder cambiar esta configuración y proceder a la actualización.

Note

- Si tanto `PlacementGroup/Enabled` como `Name` no están configurados, sus valores respectivos son los valores predeterminados de la configuración de [SlurmQueues/Networking/PlacementGroup](#).
- `ComputeResources/Networking/PlacementGroup` se añade con la AWS ParallelCluster versión 3.3.0.

CustomSlurmSettings (Opcional, **Dict**)

(Opcional) Define los ajustes de configuración del nodo de Slurm personalizado (recurso informático).

Especifica un diccionario de pares clave-valor de parámetros de configuración de Slurm personalizados que se aplican a los nodos de Slurm (recursos informáticos).

Cada par clave-valor independiente, por ejemplo `Param1: Value1`, se añade por separado al final de la línea de configuración del nodo de Slurm en el formato `Param1=Value1`.

Solo puede especificar los parámetros de configuración de Slurm que no estén incluidos en la lista de denegaciones de `CustomSlurmSettings`. Para obtener más información acerca de los


parámetros de configuración de Slurm incluidos en la lista de denegaciones, consulte [Parámetros de configuración de Slurm enumerados en denegación para CustomSlurmSettings](#).

AWS ParallelCluster solo comprueba si un parámetro está en una lista de denegaciones. AWS ParallelCluster no valida la sintaxis ni la semántica de los parámetros de Slurm configuración personalizados. Usted es responsable de validar los parámetros de configuración personalizados de Slurm. Los parámetros de configuración personalizados de Slurm no válidos pueden provocar errores en los daemons de Slurm y provocar errores en la creación y actualización del clúster.

Para obtener más información sobre cómo especificar parámetros de Slurm configuración personalizados con AWS ParallelCluster, consulte [Personalización de la configuración de Slurm](#).

Para obtener más información sobre los parámetros de configuración de Slurm, consulte [slurm.conf](#) en la documentación de Slurm.

Política de actualización: esta configuración se puede cambiar durante una actualización.

 Note

CustomSlurmSettings es compatible a partir de AWS ParallelCluster versión 3.6.0.

(Cadena) Opcional.

Una lista de pares clave-valor de etiqueta. Las etiquetas de ComputeResource anulan las etiquetas duplicadas especificadas en las [Sección de Tags](#) o [SlurmQueues/Tags](#).

Key (Opcional, **String**)

La clave de la etiqueta.

Value (Opcional, **String**)

El valor de la etiqueta.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

ComputeSettings

(Obligatorio) Define la configuración de ComputeSettings de la cola de Slurm.

Propiedades de **ComputeSettings**

Especifica las propiedades de **ComputeSettings** de los nodos de la cola de Slurm.

```
ComputeSettings:  
  LocalStorage:  
    RootVolume:  
      Size: integer  
      Encrypted: boolean  
      VolumeType: string  
      Iops: integer  
      Throughput: integer  
    EphemeralVolume:  
      MountDir: string
```

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

LocalStorage (opcional)

Especifica las propiedades de **LocalStorage** de los nodos de la cola de Slurm.

```
LocalStorage:  
  RootVolume:  
    Size: integer  
    Encrypted: boolean  
    VolumeType: string  
    Iops: integer  
    Throughput: integer  
  EphemeralVolume:  
    MountDir: string
```

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

RootVolume (opcional)

Especifica los detalles del volumen raíz de los nodos de la cola de Slurm.

```
RootVolume:  
  Size: integer  
  Encrypted: boolean  
  VolumeType: string
```

Iops: *integer*

Throughput: *integer*

Política de actualización: debe haberse detenido la flota de computación o configurado `QueueUpdateStrategy` para poder cambiar esta configuración y proceder a la actualización.

Size (Opcional, **Integer**)

Especifica el tamaño del volumen raíz en gibibytes (GiB) para los nodos de la cola de Slurm. El tamaño predeterminado proviene de la AMI. El uso de un tamaño diferente requiere que la AMI admita `growroot`.

Política de actualización: debe haberse detenido la flota de computación o configurado `QueueUpdateStrategy` para poder cambiar esta configuración y proceder a la actualización.

Encrypted (Opcional, **Boolean**)

Si `true`, el volumen raíz de los nodos de la cola de Slurm está cifrado. El valor predeterminado es `false`.

Política de actualización: debe haberse detenido la flota de computación o configurado `QueueUpdateStrategy` para poder cambiar esta configuración y proceder a la actualización.

VolumeType (Opcional, **String**)

Especifica el [tipo de volumen de Amazon EBS](#) de los nodos de la cola de Slurm. Los valores admitidos son `gp2`, `gp3`, `io1`, `io2`, `sc1`, `st1` y `standard`. El valor predeterminado es `gp3`.

Para obtener más información, consulte [Tipos de volúmenes de Amazon EBS](#) en la Guía del usuario de Amazon EC2.

Política de actualización: debe haberse detenido la flota de computación o configurado `QueueUpdateStrategy` para poder cambiar esta configuración y proceder a la actualización.

Iops (Opcional, **Boolean**)

Define el número de IOPS para volúmenes de tipo `io1`, `io2` y `gp3`.

El valor predeterminado, los valores admitidos y la proporción de datos `volume_iops` a `volume_size` varía entre `VolumeType` y `Size`.

VolumeType = io1

Iops predeterminado = 100

Valores admitidos Iops = 100–64 000 †

Relación máxima `volume_iops` a `volume_size` = 50 IOPS por GiB. 5000 IOPS requieren un `volume_size` de al menos 100 GiB.

VolumeType = io2

Iops predeterminado = 100

Valores admitidos Iops = 100–64 000 (256 000 para los volúmenes de io2 Block Express) †

Relación máxima Iops a Size = 500 IOPS por GiB. 5000 IOPS requieren un Size de al menos 10 GiB.

VolumeType = gp3

Iops predeterminado = 3000

Valores admitidos Iops = 3000–16 000 †

Proporción máxima de Iops a Size = 500 IOPS por GiB para volúmenes con IOPS superiores a 3000.

† Las IOPS máximas solo se garantizan en [las instancias creadas en el sistema Nitro](#) aprovisionadas con más de 32 000 IOPS. Otras instancias garantizan hasta 32,000 IOPS. Es posible que los volúmenes io1 más antiguos no alcancen el rendimiento máximo a menos que [modifique el volumen](#). io2 Los volúmenes de Block Express admiten valores de `volume_iops` de hasta 256 000 en tipos de instancias R5b. Para obtener más información, consulte [io2Block Express Volumes](#) en la Guía del usuario de Amazon EC2.

[Política de actualización: debe haberse detenido la flota de computación o configurado `QueueUpdateStrategy` para poder cambiar esta configuración y proceder a la actualización.](#)

Throughput (Opcional, **Integer**)

Define el rendimiento de los tipos de volumen gp3, en MiB/s. Esta configuración solo es válida cuando `VolumeType` se establece en gp3. El valor predeterminado es 125. Valores admitidos = 125–1000 MiB/s

La relación entre Throughput y Iops no puede ser superior a 0,25. El rendimiento máximo de 1000 MiB/s requiere que la configuración de Iops sea de al menos 4000.

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

EphemeralVolume (Opcional, **Boolean**)

Especifica la configuración del volumen efímero. El volumen efímero se crea combinando todos los volúmenes del almacén de instancias en un único volumen lógico formateado con el sistema de archivos ext4. El valor predeterminado es `/scratch`. Si el tipo de instancia no tiene ningún volumen de almacén de instancias, no se crea ningún volumen efímero. Para obtener más información, consulte [Volúmenes de almacén de instancias](#) en la Guía del usuario de Amazon EC2.

```
EphemeralVolume:  
MountDir: string
```

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

MountDir (Opcional, **String**)

El directorio de montaje del volumen efímero de cada nodo de la cola de Slurm.

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

CustomActions

(Opcional) Especifica los scripts personalizados que se ejecutarán en los nodos de la cola de Slurm.

```
CustomActions:  
OnNodeStart:  
  Sequence:  
    - Script: string  
      Args:  
        - string  
  Script: string  
  Args:
```



```
- string  
OnNodeConfigured:  
  Sequence:  
    - Script: string  
    Args:  
      - string  
  Script: string  
  Args:  
    - string
```

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Propiedades de **CustomActions**

OnNodeStart (Opcional, **String**)

Especifica una secuencia de scripts o un script único que se ejecutará en los nodos de la Slurm cola antes de que se inicie cualquier acción de arranque de implementación de nodos. AWS ParallelCluster no admite incluir un único script y Sequence para la misma acción personalizada. Para obtener más información, consulte [Acciones de arranque personalizadas](#).

Sequence (opcional)

Lista de scripts que se van a ejecutar.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Script (Requerido, **String**)

El archivo que se va a utilizar. La ruta del archivo debe comenzar con `https://` o `s3://`.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Args (Opcional, **[String]**)

La lista de argumentos que se pasan al script único.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Script (Requerido, **String**)

El archivo que se va a utilizar para un único script. La ruta del archivo debe comenzar con `https://` o `s3://`.

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

Args (Opcional, **[String]**)

La lista de argumentos que se pasan al script único.

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

OnNodeConfigured (Opcional, **String**)

Especifica una secuencia de scripts o un solo script para ejecutarse en los nodos de la cola de Slurm una vez completadas todas las acciones de arranque de los nodos. AWS ParallelCluster no admite incluir un único script y Sequence para la misma acción personalizada. Para obtener más información, consulte [Acciones de arranque personalizadas](#).

Sequence (opcional)

Lista de scripts que se van a ejecutar.

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

Script (Requerido, **String**)

El archivo que se va a utilizar. La ruta del archivo debe comenzar con `https://` o `s3://`.

[Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.](#)

Args (Opcional, **[String]**)

La lista de argumentos que se pasan al script único.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Script (Requerido, **String**)

El archivo que se va a utilizar para un único script. La ruta del archivo debe comenzar con `https://` o `s3://`.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Args (Opcional, [**String**])

La lista de argumentos que se pasan al script único.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Política de actualización: debe haberse detenido la flota de computación o configurado QueueUpdateStrategy para poder cambiar esta configuración y proceder a la actualización.

Note

Sequencese añade a partir de la AWS ParallelCluster versión 3.6.0. Si lo especifica `Sequence`, puede enumerar varios scripts para una acción personalizada. AWS ParallelCluster sigue admitiendo la configuración de una acción personalizada con un único script, sin incluirlo `Sequence`.

AWS ParallelCluster no admite incluir tanto un único script como `Sequence` para la misma acción personalizada.

Iam

(Opcional) Define la configuración de IAM opcional para la cola de Slurm.

```
Iam:
  S3Access:
    - BucketName: string
      EnableWriteAccess: boolean
      KeyName: string
```

AdditionalIamPolicies:

- **Policy:** *string*

InstanceProfile: *string*

InstanceRole: *string*

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

Propiedades de **Iam**

InstanceProfile (Opcional, **String**)

Especifica un perfil de instancia para anular el rol de instancia o el perfil de instancia predeterminados de la cola de Slurm. No puede especificar InstanceProfile ni InstanceRole. El formato es `arn:${Partition}:iam:${Account}:instance-profile/${InstanceProfileName}`.

Si se especifica, no se puede especificar la configuración de AdditionalIamPolicies y S3Access.

Le recomendamos que especifique una o ambas opciones de configuración de S3Access y AdditionalIamPolicies, ya que las características que se añaden a AWS ParallelCluster menudo requieren nuevos permisos.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

InstanceRole (Opcional, **String**)

Especifica un rol de instancia para anular el rol de instancia o el perfil de instancia predeterminados de la cola de Slurm. No puede especificar InstanceProfile ni InstanceRole. El formato es `arn:${Partition}:iam:${Account}:role/${RoleName}`.

Si se especifica, no se puede especificar la configuración de AdditionalIamPolicies y S3Access.

Le recomendamos que especifique una o ambas opciones de configuración de S3Access y AdditionalIamPolicies, ya que las características que se añaden a AWS ParallelCluster menudo requieren nuevos permisos.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

S3Access (opcional)

Especifica un bucket para la cola de Slurm. Se utiliza para generar políticas que concedan el acceso especificado al bucket de la cola de Slurm.

Si se especifica, no se puede especificar la configuración de InstanceRole y InstanceProfile.

Le recomendamos que especifique una o ambas opciones de configuración de S3Access y AdditionalIamPolicies, ya que las características que se añaden a AWS ParallelCluster menudo requieren nuevos permisos.

S3Access:

- BucketName: *string*
- EnableWriteAccess: *boolean*
- KeyName: *string*

Política de actualización: esta configuración se puede cambiar durante una actualización.

BucketName (Requerido, **String**)

Nombre del bucket de .

Política de actualización: esta configuración se puede cambiar durante una actualización.

KeyName (Opcional, **String**)

La clave del bucket. El valor predeterminado es *.

Política de actualización: esta configuración se puede cambiar durante una actualización.

EnableWriteAccess (Opcional, **Boolean**)

Indica si se ha habilitado el acceso a la escritura en el bucket.

Política de actualización: esta configuración se puede cambiar durante una actualización.

AdditionalIamPolicies (opcional)

Especifica una lista separada de nombres de recursos de Amazon (ARN) de políticas de IAM para Amazon EC2. Esta lista se adjunta a la función raíz utilizada para la Slurm cola, además de los permisos requeridos por AWS ParallelCluster ella.

El nombre de una política de IAM y su ARN son diferentes. Los nombres no se pueden usar.

Si se especifica, no se puede especificar la configuración de InstanceRole y InstanceProfile.

Le recomendamos que use las AdditionalIamPolicies porque se añaden AdditionalIamPolicies a los permisos que requiere AWS ParallelCluster y el InstanceRole debe incluir todos los permisos necesarios. Los permisos necesarios a menudo cambian de versión a versión a medida que se añaden características.

No hay valor predeterminado.

AdditionalIamPolicies:

- Policy: *string*

Política de actualización: esta configuración se puede cambiar durante una actualización.

Policy (Requerido, **[String]**)

Lista de políticas de IAM.

Política de actualización: esta configuración se puede cambiar durante una actualización.

SlurmSettings

(Opcional) Define la configuración de Slurm que se aplica a todo el clúster.

SlurmSettings:

ScaledownIdleTime: *integer*
QueueUpdateStrategy: *string*
EnableMemoryBasedScheduling: *boolean*
CustomSlurmSettings: *[dict]*
CustomSlurmSettingsIncludeFile: *string*
Database:
Uri: *string*
UserName: *string*
PasswordSecretArn: *string*
Dns:
DisableManagedDns: *boolean*
HostedZoneId: *string*
UseEc2Hostnames: *boolean*

Propiedades de **SlurmSettings**

ScaledownIdleTime (Opcional, **Integer**)

Define el tiempo (en minutos) durante el cual no hay trabajo y el nodo de Slurm finaliza.

El valor predeterminado es 10.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

MungeKeySecretArn (Opcional, **String**)

El nombre de recurso de Amazon (ARN) del secreto de AWS Secrets Manager en texto simple que contiene la clave munge codificada en base64 que se utilizará en el clúster de Slurm. Esta clave munge se utilizará para autenticar las llamadas RPC entre los comandos del cliente Slurm y los daemons de Slurm que actúan como servidores remotos. Si no MungeKeySecretArn se proporciona, AWS ParallelCluster generará una clave munge aleatoria para el clúster.

Note

MungeKeySecretArn se admite a partir de la AWS ParallelCluster versión 3.8.0.

Warning

Si MungeKeySecretArn se acaba de añadir a un clúster existente, no ParallelCluster restaurará la clave munge anterior en caso de una reversión o cuando la elimine más tarde. MungeKeySecretArn En su lugar, se generará una nueva clave de munge aleatoria.

Si el AWS ParallelCluster usuario tiene permiso para utilizar [DescribeSecrets](#) ese recurso secreto específico, MungeKeySecretArn se valida. MungeKeySecretArn es válido si:

- El secreto especificado existe y
- El secreto es texto plano y contiene una cadena válida codificada en base64, y
- La clave munge binaria decodificada tiene un tamaño entre 256 y 8192 bits.

Si la política de IAM del usuario de pcluster no la incluye DescribeSecret, no se valida y MungeKeySecretArn se muestra un mensaje de advertencia. Para obtener más información, consulte [Política de usuario básica pcluster de AWS ParallelCluster](#).

Al realizar la actualización `MungeKeySecretArn`, se deben detener la flota informática y todos los nodos de inicio de sesión.

Si el valor secreto del ARN secreto se modifica mientras el ARN sigue siendo el mismo, el clúster no se actualizará automáticamente con la nueva clave de munge. Para utilizar la nueva clave munge del ARN secreto, debe detener la flota de cómputo y los nodos de inicio de sesión y, a continuación, ejecutar el siguiente comando desde el nodo principal.

```
sudo /opt/parallelcluster/scripts/slurm/update_munge_key.sh
```

Tras ejecutar el comando, puede reanudar tanto la flota de procesamiento como los nodos de inicio de sesión: los nodos de procesamiento e inicio de sesión recién provisionados comenzarán a utilizar automáticamente la nueva clave munge.

Para generar una clave munge personalizada codificada en base64, puede usar la utilidad `mungekey` que se distribuye con el software [munge y, a continuación, codificarla con la utilidad base64](#) generalmente disponible en su sistema operativo. Como alternativa, puede utilizar `bash` (establezca el parámetro `bs` entre 32 y 1024)

```
dd if=/dev/random bs=128 count=1 2>/dev/null | base64 -w 0
```

o Python de la siguiente manera:

```
import random
import os
import base64

# key length in bytes
key_length=128

base64.b64encode(os.urandom(key_length)).decode("utf-8")
```

Política de actualización: SE DETUVO LA NUEVA POLÍTICA DE ACTUALIZACIÓN CON LA FLOTA DE CÓMPUTOS Y LOS NODOS DE INICIO DE SESIÓN (no se agregó por error en la versión 3.7.0).

QueueUpdateStrategy (Opcional, **String**)

Especifica la estrategia de reemplazo de los parámetros de la sección de [SlurmQueues](#) que tienen la siguiente política de actualización:

Política de actualización: debe haberse detenido la flota de computación o configurado `QueueUpdateStrategy` para poder cambiar esta configuración y proceder a la actualización.

El valor `QueueUpdateStrategy` solo se usa cuando se inicia un proceso de actualización del clúster.

Valores válidos: `COMPUTE_FLEET_STOP` | `DRAIN` | `TERMINATE`

Valor predeterminado: `COMPUTE_FLEET_STOP`

DRAIN

Los nodos de las colas con valores de parámetros cambiados se establecen en `DRAINING`. Los nodos en este estado no aceptan nuevos trabajos y los trabajos en ejecución continúan completándose.

Cuando un nodo se convierte en `idle` (`DRAINED`), se reemplaza un nodo si es estático y se termina si es dinámico. Los demás nodos de otras colas sin cambios en los valores de los parámetros no se ven afectados.

El tiempo que necesita esta estrategia para reemplazar todos los nodos de cola por valores de parámetros modificados depende de la carga de trabajo en ejecución.

COMPUTE_FLEET_STOP

El valor predeterminado del parámetro es `.`. Con esta configuración, la actualización de los parámetros de la sección [SlurmQueues](#) requiere que [detenga la flota de computación](#) antes de realizar una actualización del clúster:

```
$ pcluster update-compute-fleet --status STOP_REQUESTED
```

TERMINATE

En las colas con valores de parámetros modificados, los trabajos en ejecución finalizan y los nodos se apagan inmediatamente.

Los nodos estáticos se sustituyen y los nodos dinámicos se terminan.

Los demás nodos de otras colas sin cambios en los valores de los parámetros no se ven afectados.

Política de actualización: esta configuración no se analiza durante una actualización.

Note

QueueUpdateStrategy es compatible a partir de AWS ParallelCluster versión 3.2.0.

EnableMemoryBasedScheduling (Opcional, **Boolean**)

Si se establece en `true`, la programación basada en memoria está habilitada en Slurm. Para obtener más información, consulte [SlurmQueues](#). [ComputeResources](#) [SchedulableMemory](#).

El valor predeterminado es `false`.

Warning

La activación de la programación basada en memoria afecta a la forma en que el programador de Slurm gestiona las tareas y la asignación de nodos. Para obtener más información, consulte [Slurm programación basada en memoria](#).

Note

EnableMemoryBasedScheduling es compatible a partir de AWS ParallelCluster versión 3.2.0.

Note

[A partir de AWS ParallelCluster la versión 3.7.0, se EnableMemoryBasedScheduling puede habilitar si configuras varios tipos de instancias en Instances.](#)

Para AWS ParallelCluster las versiones 3.2.0 a 3.6. x, no se EnableMemoryBasedScheduling puede habilitar si configuras varios tipos de [instancias en Instances](#).

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

CustomSlurmSettings (Opcional, **[Dict]**)

Define la configuración de Slurm personalizada que se aplica a todo el clúster.

Especifica una lista de diccionarios de configuración de Slurm de pares clave-valor que se añadirán al final del archivo `slurm.conf` que AWS ParallelCluster genera.

Cada diccionario de la lista aparece como una línea independiente que se agrega al archivo de configuración de Slurm. Puede especificar parámetros simples o complejos.

Los parámetros simples se componen de un único key pair, tal y como se muestra en los siguientes ejemplos:

```
- Param1: 100
- Param2: "SubParam1,SubParam2=SubValue2"
```

Ejemplo renderizado en la configuración de Slurm:

```
Param1=100
Param2=SubParam1,SubParam2=SubValue2
```

Los parámetros de configuración de Slurm complejos constan de varios pares clave-valor separados por espacios, como se muestra en los siguientes ejemplos:

```
- NodeName: test-nodes[1-10]
  CPUs: 4
  RealMemory: 4196
  ... # other node settings
- NodeSet: test-nodeset
  Nodes: test-nodes[1-10]
  ... # other nodeset settings
- PartitionName: test-partition
  Nodes: test-nodeset
  ... # other partition settings
```

Ejemplo, renderizado en la configuración de Slurm:

```
NodeName=test-nodes[1-10] CPUs=4 RealMemory=4196 ... # other node settings
NodeSet=test-nodeset Nodes=test-nodes[1-10] ... # other nodeset settings
PartitionName=test-partition Nodes=test-nodeset ... # other partition settings
```

Note

Los nodos de Slurm personalizados no deben contener los patrones `-st-` `ni` `-dy-` en sus nombres. Estos patrones están reservados para los nodos administrados por AWS ParallelCluster.

Si especifica parámetros de configuración de Slurm personalizados en `CustomSlurmSettings`, no debe especificar parámetros de configuración de Slurm personalizados para `CustomSlurmSettingsIncludeFile`.

Solo puede especificar los parámetros de configuración de Slurm que no estén incluidos en la lista de denegaciones de `CustomSlurmSettings`. Para obtener más información acerca de los parámetros de configuración de Slurm incluidos en la lista de denegaciones, consulte [Parámetros de configuración de Slurm enumerados en denegación para CustomSlurmSettings](#).

AWS ParallelCluster solo comprueba si un parámetro está en una lista de denegaciones. AWS ParallelCluster no valida la sintaxis ni la semántica de los parámetros de Slurm configuración personalizados. Usted es responsable de validar los parámetros de configuración personalizados de Slurm. Los parámetros de configuración personalizados de Slurm no válidos pueden provocar errores en los daemons de Slurm y provocar errores en la creación y actualización del clúster.

Para obtener más información sobre cómo especificar parámetros de Slurm configuración personalizados con AWS ParallelCluster, consulte [Personalización de la configuración de Slurm](#).

Para obtener más información sobre los parámetros de configuración de Slurm, consulte [slurm.conf](#) en la documentación de Slurm.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

Note

`CustomSlurmSettings` es compatible a partir de AWS ParallelCluster versión 3.6.0.


CustomSlurmSettingsIncludeFile (Opcional, **String**)

Define la configuración de Slurm personalizada que se aplica a todo el clúster.

Especifica el archivo de Slurm personalizado compuesto por los parámetros de configuración de Slurm personalizados que se añadirán al final del archivo de `slurm.conf` que genera AWS ParallelCluster .

Debe incluir la ruta al archivo. La ruta del archivo debe comenzar con `https://` o `s3://`.

Si especifica parámetros de configuración de Slurm personalizados para `CustomSlurmSettingsIncludeFile`, no debe especificar parámetros de configuración de Slurm personalizados para `CustomSlurmSettings`.

 Note

Los nodos de Slurm personalizados no deben contener los patrones `-st-` `ni` `-dy-` en sus nombres. Estos patrones están reservados para los nodos administrados por AWS ParallelCluster.

Solo puede especificar los parámetros de configuración de Slurm que no estén incluidos en la lista de denegaciones de `CustomSlurmSettingsIncludeFile`. Para obtener más información acerca de los parámetros de configuración de Slurm incluidos en la lista de denegaciones, consulte [Parámetros de configuración de Slurm enumerados en denegación para CustomSlurmSettings](#).

AWS ParallelCluster comprueba únicamente si un parámetro está en una lista de denegaciones. AWS ParallelCluster no valida la sintaxis ni la semántica de los parámetros de Slurm configuración personalizados. Usted es responsable de validar los parámetros de configuración personalizados de Slurm. Los parámetros de configuración personalizados de Slurm no válidos pueden provocar errores en los daemons de Slurm y provocar errores en la creación y actualización del clúster.

Para obtener más información sobre cómo especificar parámetros de Slurm configuración personalizados con AWS ParallelCluster, consulte [Personalización de la configuración de Slurm](#).

Para obtener más información sobre los parámetros de configuración de Slurm, consulte [slurm.conf](#) en la documentación de Slurm.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

Note

CustomSlurmSettings es compatible a partir de AWS ParallelCluster versión 3.6.0.

Database

(Opcional) Define la configuración para habilitar la contabilidad de Slurm en el clúster. Para obtener más información, consulte [Slurmcontabilidad con AWS ParallelCluster](#).

Database:

Uri: *string*

UserName: *string*

PasswordSecretArn: *string*

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

Propiedades de Database

Uri (Requerido, String)

La dirección del servidor de base de datos que se utiliza como servidor de contabilidad de Slurm. Este URI debe tener el mismo formato que `host:port` y no debe contener un esquema, como `mysql://`. El host puede ser una dirección IP o un nombre DNS que pueda resolver el nodo principal. Si no se proporciona un puerto, AWS ParallelCluster utiliza el puerto predeterminado 3306 de MySQL.

AWS ParallelCluster inicia la base de datos de Slurm contabilidad al clúster y debe acceder a la base de datos.

Se debe poder acceder a la base de datos antes de que ocurra lo siguiente:

- Se crea un clúster.
- La contabilidad de Slurm se habilita con una actualización del clúster.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

UserName (Requerido, **String**)

La identidad que se Slurm utiliza para conectarse a la base de datos, escribir registros contables y realizar consultas. El usuario debe tener permisos de lectura y escritura en la base de datos.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

PasswordSecretArn (Requerido, **String**)

El nombre de recurso de Amazon (ARN) del AWS Secrets Manager secreto que contiene la contraseña en texto UserName simple. Esta contraseña se utiliza junto con la contabilidad de UserName y Slurm para autenticarse en el servidor de la base de datos.

Note

Al crear un secreto con la AWS Secrets Manager consola, asegúrese de seleccionar «Otro tipo de secreto», seleccionar texto sin formato e incluir solo el texto de la contraseña en el secreto.

Para obtener más información sobre cómo AWS Secrets Manager crear un secreto, consulta [Crear un AWS Secrets Manager secreto](#)

Si el usuario tiene el permiso para hacerlo [DescribeSecret](#), PasswordSecretArn está validado. PasswordSecretArnes válido si existe el secreto especificado. Si la política de IAM del usuario no la incluye DescribeSecret, PasswordSecretArn no está validada y aparece un mensaje de advertencia. Para obtener más información, consulte [Política de usuario básica pcluster de AWS ParallelCluster](#).

Al actualizar PasswordSecretArn, la flota de computación debe detenerse. Si el valor secreto cambia y el ARN secreto no cambia, el clúster no se actualiza automáticamente con la nueva contraseña de la base de datos. Para actualizar el clúster para el nuevo valor secreto, debe ejecutar el siguiente comando desde dentro del nodo principal después de detener la flota de computación.

```
$ sudo /opt/parallelcluster/scripts/slurm/update_slurm_database_password.sh
```

⚠ Warning

Para evitar perder datos de contabilidad, le recomendamos que cambie únicamente la contraseña de la base de datos cuando la flota de computación esté detenida.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

DatabaseName (Opcional, **String**)

Nombre de la base de datos en el servidor de bases de datos (definido por el parámetro Uri) que se utilizará para Slurm Accounting.

El nombre de la base de datos puede contener letras minúsculas, números y guiones bajos. El nombre no puede tener más de 64 caracteres.

Este parámetro se asigna al StorageLoc parámetro de [slurmdbd.conf](#).

Si no DatabaseName se proporciona, ParallelCluster utilizará el nombre del clúster para definir un valor. StorageLoc

DatabaseNameSe permite actualizar el, teniendo en cuenta las siguientes consideraciones:

- Si aún DatabaseName no existe una base de datos con un nombre en el servidor de bases de datos, slurmdbd la creará. Será su responsabilidad volver a configurar la nueva base de datos según sea necesario (por ejemplo, añadiendo las entidades contables: clústeres, cuentas, usuarios, asociaciones, QoS, etc.).
- Si DatabaseName ya existe una base de datos con un nombre en el servidor de bases de datos, slurmdbd la usará para la funcionalidad de contabilidad de Slurm.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

i Note

Database se añade a partir de versión 3.3.0.

Dns

(Opcional) Define la configuración de Slurm que se aplica a todo el clúster.

Dns:

`DisableManagedDns`: *boolean*

`HostedZoneId`: *string*

`UseEc2Hostnames`: *boolean*

Propiedades de Dns

DisableManagedDns (Opcional, **Boolean**)

Si se establece en `true`, las entradas de DNS del clúster no se crean y los nombres de los nodos de Slurm no se pueden resolver.

De forma predeterminada, AWS ParallelCluster crea una zona alojada en Route 53 en la que los nodos se registran cuando se lanzan. El valor predeterminado es `false`. Si `DisableManagedDns` se establece en `true`, la zona alojada no la creó AWS ParallelCluster.

Para obtener información sobre cómo usar esta configuración para implementar clústeres en subredes sin acceso a Internet, consulte [AWS ParallelCluster en una subred individual sin acceso a Internet](#).

Warning

Se requiere un sistema de resolución de nombres para que el clúster funcione correctamente. Si `DisableManagedDns` está establecido en `true`, debe proporcionar un sistema de resolución de nombres. Para usar el DNS predeterminado de EC2, establezca `UseEc2Hostnames` en `true`. Como alternativa, configure su propia resolución de DNS y asegúrese de que los nombres de los nodos estén registrados cuando se lancen las instancias. Por ejemplo, puede hacerlo configurando [CustomActions/OnNodeStart](#).

Política de actualización: si se cambia esta configuración, no se permite la actualización.

HostedZoneId (Opcional, **String**)

Define un ID de zona alojada de Route 53 personalizado para usarlo en la resolución de nombres de DNS del clúster. Cuando se proporciona, AWS ParallelCluster registra los nodos del clúster en la zona alojada especificada y no crea una zona alojada administrada.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

UseEc2Hostnames (Opcional, **Boolean**)

Si se establece en `true`, los nodos de cómputo del clúster están configurados con el nombre de host EC2 predeterminado. El `NodeHostName` de Slurm también se actualiza con esta información. El valor predeterminado es `false`.

Para obtener información sobre cómo usar esta configuración para implementar clústeres en subredes sin acceso a Internet, consulte [AWS ParallelCluster en una subred individual sin acceso a Internet](#).

Note

Esta nota no es pertinente a partir de AWS ParallelCluster versión 3.3.0.

Para las versiones AWS ParallelCluster compatibles anteriores a la 3.3.0:

Cuando `UseEc2Hostnames` se establece en `true`, el archivo de configuración de Slurm se establece con los AWS ParallelCluster prolog scripts y: `epilog`

- `prolog` se ejecuta para añadir información de los nodos a `/etc/hosts` sobre los nodos de computación cuando se asigna cada trabajo.
- `epilog` se ejecuta para limpiar el contenido escrito por `prolog`.

Para añadir los scripts `prolog` o `epilog` personalizados, agréguelos a las carpetas `/opt/slurm/etc/pcluster/prolog.d/` o `/opt/slurm/etc/pcluster/epilog.d/`, respectivamente.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Sección de **SharedStorage**

(Opcional) La configuración de almacenamiento compartido del clúster.


AWS ParallelCluster [admite el uso de Amazon EBS, FSx para ONTAP y FSx para los volúmenes de almacenamiento compartido OpenZFS, Amazon EFS y FSx para los sistemas de archivos de almacenamiento compartido Lustre o caché de archivos](#).

En la sección de `SharedStorage`, puede definir el almacenamiento externo o administrado:

- El almacenamiento externo se refiere a un volumen o sistema de archivos existente que usted administra. AWS ParallelCluster no lo crea ni lo elimina.

- AWS ParallelCluster el almacenamiento gestionado se refiere a un volumen o sistema de archivos que AWS ParallelCluster creó y puede eliminar.

Para ver [las cuotas de almacenamiento compartido](#) y obtener más información sobre la configuración del almacenamiento compartido, consulte [Almacenamiento compartido](#) Uso de AWS ParallelCluster.

 Note

Si AWS Batch se utiliza como planificador, FSx for Lustre solo está disponible en el nodo principal del clúster.

SharedStorage:

- MountDir: *string*
Name: *string*
StorageType: Ebs
EbsSettings:
 - VolumeType: *string*
 - Iops: *integer*
 - Size: *integer*
 - Encrypted: *boolean*
 - KmsKeyId: *string*
 - SnapshotId: *string*
 - Throughput: *integer*
 - VolumeId: *string*
 - DeletionPolicy: *string*
 - Raid:
 - Type: *string*
 - NumberOfVolumes: *integer*
- MountDir: *string*
Name: *string*
StorageType: Efs
EfsSettings:
 - Encrypted: *boolean*
 - KmsKeyId: *string*
 - EncryptionInTransit: *boolean*
 - IamAuthorization: *boolean*
 - PerformanceMode: *string*
 - ThroughputMode: *string*
 - ProvisionedThroughput: *integer*
 - FileSystemId: *string*

- DeletionPolicy: *string*
- MountDir: *string*
 - Name: *string*
 - StorageType: FsxLustre
 - FsxLustreSettings:
 - StorageCapacity: *integer*
 - DeploymentType: *string*
 - ImportedFileChunkSize: *integer*
 - DataCompressionType: *string*
 - ExportPath: *string*
 - ImportPath: *string*
 - WeeklyMaintenanceStartTime: *string*
 - AutomaticBackupRetentionDays: *integer*
 - CopyTagsToBackups: *boolean*
 - DailyAutomaticBackupStartTime: *string*
 - PerUnitStorageThroughput: *integer*
 - BackupId: *string*
 - KmsKeyId: *string*
 - FileSystemId: *string*
 - AutoImportPolicy: *string*
 - DriveCacheType: *string*
 - StorageType: *string*
 - DeletionPolicy: *string*
 - DataRepositoryAssociations:
 - Name: *string*
 - BatchImportMetadataOnCreate: *boolean*
 - DataRepositoryPath: *string*
 - FileSystemPath: *string*
 - ImportedFileChunkSize: *integer*
 - AutoExportPolicy: *string*
 - AutoImportPolicy: *string*
- MountDir: *string*
 - Name: *string*
 - StorageType: FsxOntap
 - FsxOntapSettings:
 - VolumeId: *string*
- MountDir: *string*
 - Name: *string*
 - StorageType: FsxOpenZfs
 - FsxOpenZfsSettings:
 - VolumeId: *string*
- MountDir: *string*
 - Name: *string*
 - StorageType: FileCache

FileCacheSettings:**FileCacheId:** *string***SharedStorage** actualizar las políticas

- Para EBS gestionados/externos, EFS gestionados y FSx Lustre gestionados, la política de actualización es [Política de actualización: para configurar esta lista de valores, debe detenerse la flota de computación o configurarse QueueUpdateStrategy para añadir un nuevo valor; la flota de computación debe detenerse al eliminar un valor existente.](#)
- Para EFS externos, FSx Lustre, FSx ONTAP, OpenZfs FSx y caché de archivos, la política de actualización es: [Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

Propiedades de **SharedStorage**


MountDir (Requerido, String)

La ruta en la que se monta el almacenamiento compartido.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Name (Requerido, String)

El nombre del almacenamiento compartido. Este nombre se utiliza al actualizar la configuración.

 Warning

Si especifica el almacenamiento compartido AWS ParallelCluster administrado y cambia el valor, el almacenamiento compartido administrado existente y los datos se eliminan y se crea un nuevo almacenamiento compartido administrado. Name Cambiar el valor de Name mediante una actualización de clúster equivale a sustituir el almacenamiento compartido administrado existente por uno nuevo. Asegúrese de hacer una copia de seguridad de los datos antes de cambiarlos Name si necesita retener los datos del almacenamiento compartido existente.

[Política de actualización: para configurar esta lista de valores, debe detenerse la flota de computación o configurarse QueueUpdateStrategy para añadir un nuevo valor; la flota de computación debe detenerse al eliminar un valor existente.](#)

StorageType (Requerido, String)

El tipo de almacenamiento compartido. Los valores admitidos son Ebs, Efs, FsxLustre, FsxOntap y FsxOpenZfs.

Para obtener más información, consulte [FsxLustreSettings](#), [FsxOntapSettings](#) y [FsxOpenZfsSettings](#).

Note

Si lo utiliza AWS Batch como planificador, FSx for Lustre solo está disponible en el nodo principal del clúster.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

EbsSettings

(Opcional) La configuración de un volumen de Amazon EBS.

EbsSettings:

```
VolumeType: string  
Iops: integer  
Size: integer  
Encrypted: boolean  
KmsKeyId: string  
SnapshotId: string  
VolumeId: string  
Throughput: integer  
DeletionPolicy: string  
Raid:  
  Type: string  
  NumberOfVolumes: integer
```

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Propiedades de EbsSettings

Cuando [DeletionPolicy](#) se establece en `Delete`, un volumen gestionado, con sus datos, se elimina si se elimina el clúster o si se elimina el volumen con una actualización del clúster.

Para obtener más información, consulte [Almacenamiento compartido](#) en Uso de AWS ParallelCluster.

VolumeType (Opcional, String)

Especifica el tipo de [volumen de Amazon EBS](#). Los valores admitidos son gp2, gp3, io1, io2, sc1, st1 y standard. El valor predeterminado es gp3.

Para obtener más información, consulte [Tipos de volúmenes de Amazon EBS](#) en la Guía del usuario de Amazon EC2.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Iops (Opcional, Integer)

Define el número de IOPS para volúmenes de tipo io1, io2 y gp3.

El valor predeterminado, los valores admitidos y la proporción de datos `volume_iops` a `volume_size` varía entre VolumeType y Size.

VolumeType = io1

Iops predeterminado = 100

Valores admitidos Iops = 100–64 000 †

Relación máxima `volume_iops` a `volume_size` = 50 IOPS por cada GiB. 5000 IOPS requieren un `volume_size` de al menos 100 GiB.

VolumeType = io2

Iops predeterminado = 100

Valores admitidos Iops = 100–64 000 (256 000 para los volúmenes de io2 Block Express) †

Relación máxima Iops a Size = 500 IOPS por cada GiB. 5000 IOPS requieren un Size de al menos 10 GiB.

VolumeType = gp3

Iops predeterminado = 3000

Valores admitidos Iops = 3000–16 000

Relación máxima Iops a Size = 500 IOPS por cada GiB. 5000 IOPS requieren un Size de al menos 10 GiB.

† Las IOPS máximas solo se garantizan en [las instancias creadas en el sistema Nitro](#) aprovisionadas con más de 32 000 IOPS. Otras instancias garantizan hasta 32,000 IOPS. Es posible que los volúmenes [más antiguos no alcancen el rendimiento máximo a menos que](#) modifique el volumen `io1`. `io2` Los volúmenes de Block Express admiten valores de `volume_iops` de hasta 256 000 en tipos de instancias R5b. Para obtener más información, consulte [io2Block Express Volumes](#) en la Guía del usuario de Amazon EC2.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

Size (Opcional, Integer)

El tamaño de volumen, en gibibytes (GiB). El valor predeterminado es 35.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Encrypted (Opcional, Boolean)

Especifica si el volumen de está cifrado. El valor predeterminado es `true`.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

KmsKeyId (Opcional, String)

Especifica una AWS KMS clave personalizada que se utilizará para el cifrado. Esta configuración requiere que la configuración de Encrypted esté establecida en `true`.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

SnapshotId (Opcional, String)

(Opcional) Define el ID de instantánea de Amazon EBS si utiliza una instantánea como origen del volumen.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

VolumeId (Opcional, String)

Especifica el ID de volumen de Amazon EBS. Cuando se especifica para una instancia de `EbsSettings`, solo se puede especificar el parámetro `MountDir`.

Debe crear el volumen en la misma zona de disponibilidad que la instancia.

Note

En la AWS ParallelCluster versión 3.4.0 se añaden varias zonas de disponibilidad.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Throughput (Opcional, Integer)

El rendimiento, en MiB/s para aprovisionar un volumen, con un máximo de 1000 MiB/s.

Esta configuración solo es válida cuando `VolumeType` se establece en `gp3`. El rango admitido es de 125 a 1000, con un valor predeterminado de 125.

Política de actualización: esta configuración se puede cambiar durante una actualización.

DeletionPolicy (Opcional, String)

Especifica si el volumen debe retenerse, eliminarse o tomarse una instantánea al eliminar el clúster o eliminar el volumen. Los valores admitidos son `Delete`, `Retain` y `Snapshot`. El valor predeterminado es `Delete`.

Si se [DeletionPolicy](#) establece en `Delete`, se elimina un volumen gestionado con sus datos si se elimina el clúster o si se elimina el volumen con una actualización del clúster.

Para obtener más información, consulte [Almacenamiento compartido](#).

Política de actualización: esta configuración se puede cambiar durante una actualización.

Note

`DeletionPolicy` se admite a partir de la AWS ParallelCluster versión 3.2.0.

Raid

(Opcional) Define la configuración de un volumen RAID.

Raid:

`Type`: *string*

`NumberOfVolumes`: *integer*

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Propiedades de **Raid**

Type (Requerido, String)

Define el tipo de matriz RAID. Los valores admitidos son “0” (a rayas) y “1” (reflejado).

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

NumberOfVolumes (Opcional, Integer)

Define el número de volúmenes de Amazon EBS que se utilizarán para crear el arreglo de RAID. El rango de valores admitido es de 2 a 5. El valor predeterminado (cuando se define la configuración de Raid) es 2.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

EfsSettings

(Opcional) La configuración de un sistema de archivos de Amazon EFS.

```
EfsSettings:  
Encrypted: boolean  
KmsKeyId: string  
EncryptionInTransit: boolean  
IamAuthorization: boolean  
PerformanceMode: string  
ThroughputMode: string  
ProvisionedThroughput: integer  
FileSystemId: string  
DeletionPolicy: string
```

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Propiedades de **EfsSettings**

Si se [DeletionPolicy](#) establece en `Delete`, un sistema de archivos gestionado, con sus datos, se elimina si se elimina el clúster o si el sistema de archivos se elimina con una actualización del clúster.

Para obtener más información, consulte [Almacenamiento compartido](#) en Uso de AWS ParallelCluster.

Encrypted (Opcional, Boolean)

Especifica si el sistema de archivos está cifrado. El valor predeterminado es `false`.

Política de actualización: si se cambia esta configuración, no se permite la actualización.


`KmsKeyId` (Opcional, `String`)

Especifica una AWS KMS clave personalizada que se utilizará para el cifrado. Esta configuración requiere que la configuración de `Encrypted` esté establecida en `true`.


Política de actualización: si se cambia esta configuración, no se permite la actualización.

`EncryptionInTransit` (Opcional, `Boolean`)

Si se establece en `true`, los sistemas de archivos de Amazon EFS se montan con seguridad de la capa de transporte (TLS). De forma predeterminada, se establece en `false`.

 Note

Si AWS Batch se usa como programador, `EncryptionInTransit` no es compatible.


 Note

`EncryptionInTransit` se añade a partir de AWS ParallelCluster versión 3.4.0.


Política de actualización: si se cambia esta configuración, no se permite la actualización.

`IamAuthorization` (Opcional, `Boolean`)

Si se establece en `true`, Amazon EFS se autentica mediante la identidad de IAM del sistema. De forma predeterminada, se establece en `false`.

 Note

Si `IamAuthorization` se establece en `true`, `EncryptionInTransit` también se debe establecer en `true`.

 Note

Si AWS Batch se usa como programador, `IamAuthorization` no es compatible.

Note

IamAuthorizationse añade a partir de la AWS ParallelCluster versión 3.4.0.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

PerformanceMode (Opcional, String)

El modo de rendimiento del sistema de archivos Amazon EFS de copia de seguridad Los valores admitidos son `generalPurpose` y `maxIO`. El valor predeterminado es `generalPurpose`. Para obtener más información, consulte [Performance modes](#) (Modos de rendimiento) en la Guía del usuario de Amazon Elastic File System.

Recomendamos el modo de rendimiento `generalPurpose` para la mayoría de sistemas de archivos.

Los sistemas de archivos que usan el modo de rendimiento de `maxIO` pueden escalar a mayores niveles de rendimiento de agregación y operaciones por segundo. Sin embargo, hay una compensación de latencias ligeramente más altas para la mayoría de las operaciones de archivos.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

ThroughputMode (Opcional, String)

Especifica el modo de rendimiento del sistema de archivos de Amazon EFS. Los valores admitidos son `bursting` y `provisioned`. El valor predeterminado es `bursting`. Cuando se usa `provisioned`, se debe especificar `ProvisionedThroughput`.

Política de actualización: esta configuración se puede cambiar durante una actualización.

ProvisionedThroughputObligatorio cuando es .

Define el rendimiento aprovisionado del sistema de archivos de Amazon EFS, medido en MiB/s. Esto corresponde al [ProvisionedThroughputInMibps](#) parámetro de la referencia de la API de Amazon EFS.

Si usa este parámetro, debe establecer `ThroughputMode` en `provisioned`.

El rango admitido es de 1 a 1024. Para solicitar un incremento del límite, póngase en contacto con AWS Support.

Política de actualización: esta configuración se puede cambiar durante una actualización.

FileSystemId (Opcional, String)

(Opcional) Define el ID del sistema de archivos de Amazon EFS para un sistema de archivos ya existente.

Si el clúster está configurado para abarcar varias zonas de disponibilidad, debe definir un objetivo de montaje del sistema de archivos en cada zona de disponibilidad que utilice el clúster.

No se puede especificar cuando se especifica `MountDir`. No se puede especificar ninguna otra `EfsSettings`.

Si establece esta opción, lo siguiente debe ser verdadero para los sistemas de archivos que defina:

- Los sistemas de archivos cuentan con un destino de montaje ya existente en cada una de las zonas de disponibilidad del clúster, con tráfico NFS de entrada y salida permitido desde `HeadNode` y `ComputeNodes`. Se configuran varias zonas de disponibilidad en [Scheduling SlurmQueues//Networking/SubnetIds](#).

Para asegurarse de que se permite el tráfico entre el clúster y el sistema de archivos, puede hacer una de las siguientes acciones:

- Configure los grupos de seguridad del destino de montaje para permitir el tráfico hacia y desde el CIDR o la lista de prefijos de las subredes del clúster.

Note

AWS ParallelCluster valida que los puertos estén abiertos y que el CIDR o la lista de prefijos estén configurados. AWS ParallelCluster no valida el contenido de la lista de bloques o prefijos del CIDR.

- Configure grupos de seguridad personalizados para los nodos del clúster mediante [SlurmQueues/Networking/SecurityGroups](#) y [HeadNode/Networking/SecurityGroups](#). Los grupos de seguridad personalizados deben configurarse para permitir el tráfico entre el clúster y el sistema de archivos.

Note

Si todos los nodos del clúster utilizan grupos de seguridad personalizados, AWS ParallelCluster solo valida que los puertos estén abiertos. AWS ParallelCluster no valida que el origen y el destino estén configurados correctamente.

Warning

EFS solo OneZone se admite si todos los nodos de procesamiento y el nodo principal se encuentran en la misma zona de disponibilidad. EFS solo OneZone puede tener un objetivo de montaje.

Note

En la AWS ParallelCluster versión 3.4.0 se añaden varias zonas de disponibilidad.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

DeletionPolicy (Opcional, String)

Especifica si el sistema de archivos debe retenerse o eliminarse al quitarlo del clúster o al eliminar el clúster. Los valores admitidos son `Delete` y `Retain`. El valor predeterminado es `Delete`.

Cuando [DeletionPolicy](#) se establece en `Delete`, un sistema de archivos gestionado, con sus datos, se elimina si se elimina el clúster o si el sistema de archivos se elimina con una actualización del clúster.

Para obtener más información, consulte [Almacenamiento compartido](#).

Política de actualización: esta configuración se puede cambiar durante una actualización.

Note

`DeletionPolicy` admite a partir de la AWS ParallelCluster versión 3.3.0.

FsxLustreSettings

Note

Debe definir FsxLustreSettings si se ha especificado FsxLustre para [StorageType](#).

(Opcional) La configuración de un sistema de archivos de FSx para Lustre.

FsxLustreSettings:

```
StorageCapacity: integer
DeploymentType: string
ImportedFileChunkSize: integer
DataCompressionType: string
ExportPath: string
ImportPath: string
WeeklyMaintenanceStartTime: string
AutomaticBackupRetentionDays: integer
CopyTagsToBackups: boolean
DailyAutomaticBackupStartTime: string
PerUnitStorageThroughput: integer
BackupId: string # BackupId cannot coexist with some of the fields
KmsKeyId: string
FileSystemId: string # FileSystemId cannot coexist with other fields
AutoImportPolicy: string
DriveCacheType: string
StorageType: string
DeletionPolicy: string
```

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Note

Si AWS Batch se utiliza como planificador, FSx for Lustre solo está disponible en el nodo principal del clúster.

Propiedades de FsxLustreSettings

Si [DeletionPolicy](#) se establece en `Delete`, un sistema de archivos gestionado, con sus datos, se elimina si se elimina el clúster o si el sistema de archivos se elimina con una actualización del clúster.

Para obtener más información, consulte [Almacenamiento compartido](#).

StorageCapacity (Requerido, Integer)

Establece la capacidad de almacenamiento del sistema de archivos que está creando.

StorageCapacity es necesario si está creando un nuevo sistema de archivos. No incluya StorageCapacity si se especifica BackupId o FileSystemId.

- Para los tipos de implementación SCRATCH_2 y , los valores válidos son 1200 GiB, 2400 GiB e incrementos de 2400 GiB.
- Para el tipo de implementación SCRATCH_1, los valores válidos son 1200 GiB, 2400 GiB e incrementos de 3600 GiB.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

DeploymentType (Opcional, String)

Especifica el tipo de implementación del sistema de archivos de FSx para Lustre. Los valores admitidos son SCRATCH_1, SCRATCH_2, PERSISTENT_1 y PERSISTENT_2. El valor predeterminado es SCRATCH_2.


Elija tipos de implementación SCRATCH_1 y SCRATCH_2 cuando necesite almacenamiento temporal y procesamiento de datos a corto plazo. El tipo de implementación SCRATCH_2 proporciona cifrado en tránsito de datos y mayor capacidad de rendimiento de ráfagas que SCRATCH_1.

Elija el tipo de implementación de PERSISTENT_1 para un almacenamiento a largo plazo y para cargas de trabajo centradas en el rendimiento que no sean sensibles a la latencia. PERSISTENT_1 admite el cifrado de los datos en tránsito. Está disponible en todos los Regiones de AWS lugares donde FSx for Lustre esté disponible.

Elija el tipo de implementación PERSISTENT_2 para almacenamiento a largo plazo y cargas de trabajo sensibles a la latencia que requieren los niveles más altos de IOPs/rendimiento. PERSISTENT_2 admite almacenamiento SSD y ofrece PerUnitStorageThroughput (hasta 1000 MB/s/Tib). PERSISTENT_2 está disponible en un número limitado de Regiones de AWS. Para obtener más información sobre los tipos de implementación y la lista de ubicaciones Regiones de AWS disponibles, consulte [las opciones de implementación del sistema de archivos para FSx for Lustre](#) en la Guía del usuario de Amazon FSx for Lustre. PERSISTENT_2

El cifrado de datos en tránsito se habilita automáticamente cuando accede a un sistema de archivos SCRATCH_2 o PERSISTENT_1 desde instancias de Amazon EC2 que PERSISTENT_2 admiten esta característica.

Se admite el cifrado de los datos en tránsito para los tipos de implementación SCRATCH_2 y , y cuando se accede desde los tipos de instancia compatibles en las regiones de AWS compatibles. Para obtener más información, consulte [Cifrado de datos en tránsito](#) en la Guía del usuario de Amazon FSx para Lustre.

 Note


La compatibilidad con este tipo de implementación de PERSISTENT_2 se agregó con la versión 3.2.0 de AWS ParallelCluster .

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

ImportedFileChunkSize (Opcional, Integer)

Para los archivos que se importan desde un repositorio de datos, este valor determina la cantidad de fragmentos y la cantidad máxima de datos por archivo (en MiB) que se almacenan en un único disco físico. El número máximo de discos en los que un único archivo se puede fraccionar está limitado por el número total de los discos que forman el sistema de archivos.

El valor predeterminado de fragmento es 1024 MiB (1 GiB) y puede llegar hasta 512 000 MiB (500 GiB). Los objetos de Amazon S3 tienen un tamaño máximo de 5 TB.

 Note

Este parámetro no se admite en sistemas de archivos que utilizan el tipo de implementación PERSISTENT_2. Para obtener instrucciones sobre cómo configurar las asociaciones de repositorios de datos, consulte [Linking your file system to an S3 bucket](#) en la Guía del usuario de Amazon FSx para Lustre.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

DataCompressionType (Opcional, String)

La configuración del sistema de archivos de Amazon FSx para Lustre. El valor admitido es LZ4. LZ4 indica que la compresión de datos está activada con el algoritmo LZ4. Si no se especifica

`DataCompressionType`, la compresión de datos se desactiva cuando se crea el sistema de archivos.

Para obtener más información, consulte [Compresión de datos de Lustre](#).

Política de actualización: esta configuración se puede cambiar durante una actualización.

ExportPath (Opcional, String)

(Opcional) La ruta en Amazon S3 a donde se exporta la raíz del sistema de archivos de Amazon FSx. Este parámetro solo se admite cuando el parámetro se establece en `.`. La ruta debe utilizar el mismo bucket de Amazon S3 tal y como se especifica en `ImportPath`. Puede proporcionar un prefijo opcional a donde se van a exportar datos nuevos y modificados desde su sistema de archivos de Amazon FSx para Lustre. Si no se indica un valor `ExportPath`, Amazon FSx establece una ruta de exportación predeterminada: `s3://import-bucket/FSxLustre[creation-timestamp]`. La marca temporal está en formato UTC. Por ejemplo `s3://import-bucket/FSxLustre20181105T222312Z`.

El bucket de exportación de Amazon S3 debe ser el mismo que el bucket de importación que especifica `ImportPath`. Si solo especifica un nombre de bucket como, por ejemplo, `s3://import-bucket`, obtendrá una asignación de objetos del sistema de archivos 1:1 para los objetos del bucket de Amazon S3. Esta asignación significa que los datos de entrada en Amazon S3 se sobrescriben al exportarse. Si proporciona un prefijo personalizado en la ruta de exportación, como, por ejemplo `s3://import-bucket/[custom-optional-prefix]`, Amazon FSx exporta el contenido de su sistema de archivos a ese prefijo de exportación en el bucket de Amazon S3.

Note

Este parámetro no se admite en sistemas de archivos que utilizan el tipo de implementación `PERSISTENT_2`. Configure las asociaciones de repositorios de datos tal y como se describe en [Linking your file system to an S3 bucket](#) de la Guía del usuario de Amazon FSx para Lustre.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

ImportPath (Opcional, String)

La ruta al bucket de Amazon S3 (incluido el prefijo opcional) que se está utilizando como repositorio de datos para el sistema de archivos de Amazon FSx para Lustre. La raíz del sistema

de archivos de FSx para Lustre se asignará a la raíz del bucket de Amazon S3 que seleccione. Un ejemplo es `s3://import-bucket/optional-prefix`. Si especifica un prefijo después del nombre del bucket de Amazon S3, solo se cargan en el sistema de archivos claves de objetos con ese prefijo.

Note

Este parámetro no se admite en sistemas de archivos que utilizan el tipo de implementación `PERSISTENT_2`. Configure las asociaciones de repositorios de datos tal y como se describe en [Linking your file system to an S3 bucket](#) de la Guía del usuario de Amazon FSx para Lustre.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

`WeeklyMaintenanceStartTime` (Opcional, String)

La hora de inicio preferida para realizar el mantenimiento semanal. Tiene el formato "`d:HH:MM`" en la zona horaria UTC+0. Para este formato, `d` es el número del día de la semana del 1 al 7, empezando por el lunes y terminando con el domingo. Se requieren comillas para este campo.

Política de actualización: esta configuración se puede cambiar durante una actualización.

`AutomaticBackupRetentionDays` (Opcional, Integer)

El número de días que se conservan las copias de seguridad automáticas. Al establecer este valor en 0, se deshabilitan las copias de seguridad automáticas. El rango admitido es de 0 a 90. El valor predeterminado es 0. Solo es válido para su uso con tipos de implementación . Para obtener más información, consulte [Trabajar con copias de seguridad](#) en la Guía del usuario de Amazon FSx for Lustre.

Política de actualización: esta configuración se puede cambiar durante una actualización.

`CopyTagsToBackups` (Opcional, Boolean)

Si es `true`, copie las etiquetas del sistema de archivos de FSx para Lustre a las copias de seguridad. Este valor se establece de forma predeterminada en `false`. Si se establece en `true`, todas las etiquetas del sistema de archivos se copian a todas las copias de seguridad automáticas e iniciadas por el usuario cuando el usuario no especifica etiquetas. Si este valor es `true` y se especifican una o varias etiquetas, solo se copian las etiquetas especificadas en las copias de seguridad. Si especifica una o varias etiquetas al crear una copia de seguridad iniciada

por el usuario, no se copiará ninguna etiqueta del sistema de archivos independientemente de ese valor. Solo es válido para su uso con tipos de implementación .

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

DailyAutomaticBackupStartTime (Opcional, String)

Un periodo de tiempo diario recurrente con el formato HH:MM. HH es la hora de relleno cero del día (0-23), y MM es el minuto de relleno cero de la hora. Por ejemplo, 05:00 especifica las 5 AM diarias. Solo es válido para su uso con tipos de implementación .

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

PerUnitStorageThroughput(Necesario para **PERSISTENT_1** los tipos de **PERSISTENT_2** implementación,Integer)

Describe el rendimiento de lectura y escritura por cada 1 tebibyte de almacenamiento, en MB/s/TiB. La capacidad de rendimiento del sistema de archivos se calcula multiplicando la capacidad de almacenamiento del sistema (TiB) por PerUnitStorageThroughput (MB/s/TiB). Para un sistema de almacenamiento de 2,4 TiB, el aprovisionamiento de 50 MB/s/TiB PerUnitStorageThroughput produce una velocidad del sistema de archivos de 120 MB/s. Usted paga la cantidad de rendimiento aprovisionada. Esto corresponde a la propiedad.

[PerUnitStorageThroughput](#)

Valores válidos:

Almacenamiento SSD PERSISTENT_1: 50, 100, 200 MB/TiB.

Almacenamiento HDD PERSISTENT_1: 12, 40 MB/TiB.

Almacenamiento SSD PERSISTENT_2: 125, 250, 1000 MB/TiB.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

BackupId (Opcional, String)

Especifica el ID de la copia de seguridad que se utilizará para restaurar el sistema de archivos FSx para Lustre a partir de una copia de seguridad existente. Cuando se especifica el ajuste BackupId, no se deben especificar los ajustes AutoImportPolicy, DeploymentType, ExportPath, KmsKeyId, ImportPath, ImportedFileChunkSize, StorageCapacity y PerUnitStorageThroughput. Estos ajustes se leen de la copia de seguridad. Además, no se deben especificar los ajustes AutoImportPolicy, ExportPath, ImportPath ni ImportedFileChunkSize. Esto corresponde a la [BackupId](#) propiedad.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

KmsKeyId (Opcional, String)


El ID de la clave AWS Key Management Service (AWS KMS) que se utiliza para cifrar los datos del sistema de archivos FSx for Lustre para los sistemas de archivos FSx for Lustre persistentes en reposo. Si no se especifica, se usa la clave administrada por FSx para Lustre. Los sistemas de archivos de Amazon FSx para Lustre SCRATCH_1 y SCRATCH_2 siempre se cifran en reposo mediante claves administradas de FSx para Lustre. Para obtener más información consulte en la Referencia de la API de .

Política de actualización: si se cambia esta configuración, no se permite la actualización.


FileSystemId (Opcional, String)

Especifica el ID de un sistema de archivos de FSx para Lustre existente.

Si se especifica esta opción, solo se utilizan los ajustes MountDir y FileSystemId en la FsxLustreSettings. Se ignoran todos los demás ajustes de FsxLustreSettings.

 Note

Si se utiliza el AWS Batch programador, FSx for Lustre solo está disponible en el nodo principal.

 Note

Si se usa un sistema de archivos ya existente, debe asociarse a un grupo de seguridad que permita el tráfico TCP de entrada y salida desde a través del puerto .

Asegúrese de que se permite el tráfico entre el clúster y el sistema de archivos de la siguiente manera:

- Configure los grupos de seguridad del sistema de archivos para permitir el tráfico hacia y desde el CIDR o la lista de prefijos de las subredes del clúster.

Note

AWS ParallelCluster valida que los puertos estén abiertos y que el CIDR o la lista de prefijos estén configurados. AWS ParallelCluster no valida el contenido de la lista de bloques o prefijos del CIDR.

- Configure grupos de seguridad personalizados para los nodos del clúster mediante [SlurmQueues/Networking/SecurityGroups](#) y [HeadNode/Networking/SecurityGroups](#). Los grupos de seguridad personalizados deben configurarse para permitir el tráfico entre el clúster y el sistema de archivos.

Note

Si todos los nodos del clúster utilizan grupos de seguridad personalizados, AWS ParallelCluster solo valida que los puertos estén abiertos. AWS ParallelCluster no valida que el origen y el destino estén configurados correctamente.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

AutoImportPolicy (Opcional, String)

Al crear el sistema de archivos, los objetos de S3 existentes aparecen como descripciones de archivos y directorios. Utilice esta propiedad para elegir cómo mantiene actualizados Amazon FSx las descripciones de archivos y directorios a medida que agrega o modifica objetos en el bucket de S3 vinculado. AutoImportPolicy puede tener los siguientes valores:

- **NEW**: la importación automática está activada. Amazon FSx importa automáticamente descripciones de directorio de cualquier objeto nuevo agregado al bucket de S3 vinculado que no exista actualmente en el sistema de archivos FSx.
- **NEW_CHANGED**: la importación automática está activada. Amazon FSx importa automáticamente descripciones de archivos y directorios de cualquier objeto nuevo agregado al bucket de S3 y de cualquier objeto existente que se cambie en el bucket de S3 después de elegir esta opción.
- **NEW_CHANGED_DELETED**: la importación automática está activada. Amazon FSx importa automáticamente listas de archivos y directorios de cualquier objeto nuevo agregado al bucket de S3, cualquier objeto existente que se modifique en el bucket de S3 y cualquier objeto que se haya eliminado en el bucket de S3.

Note

Se ha agregado compatibilidad para `NEW_CHANGED_DELETED` en la versión 3.1.1 de AWS ParallelCluster .

Si no se especifica `AutoImportPolicy`, la importación automática está desactivada. FSx para Lustre solo actualiza las descripciones de archivos y directorios desde el bucket de S3 vinculado cuando se crea el sistema de archivos. FSx no actualiza las descripciones de archivos y directorios para los objetos nuevos o modificados después de elegir esta opción.

Para obtener más información, consulte [Importar automáticamente actualizaciones desde su bucket de S3](#) en la Guía del usuario de Amazon FSx para Lustre.

Note

Este parámetro no se admite en sistemas de archivos que utilizan el tipo de implementación `PERSISTENT_2`. Para obtener instrucciones sobre cómo configurar las asociaciones de repositorios de datos, consulte [Linking your file system to an S3 bucket](#) en la Guía del usuario de Amazon FSx para Lustre.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

DriveCacheType (Opcional, String)

Especifica que el sistema de archivos tiene una memoria caché de unidad SSD. Esto solo se puede configurar si la configuración de `StorageType` está establecida en `HDD` y la configuración de `DeploymentType` está establecida en `PERSISTENT_1`. Esto corresponde a la propiedad [DriveCacheType](#). Para obtener más información, consulte [Opciones de implementación de FSx para Lustre](#) en la Guía del usuario de Amazon FSx para Lustre.

El único valor válido es `READ`. Para deshabilitar la memoria caché de la unidad SSD, no especifique la configuración de `DriveCacheType`.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

StorageType (Opcional, String)

Establece el tipo de almacenamiento para el sistema de archivos que está creando. Los valores válidos son `SSD` y `HDD`.

- Establezca en SSD para utilizar el almacenamiento de unidades de estado sólido.
- Elija HDD para utilizar el almacenamiento de la unidad de disco duro. HDD es compatible con los tipos de implementación de PERSISTENT.

El valor predeterminado es SSD. Para obtener más información, consulte [Opciones de tipo de almacenamiento](#) en la Guía del usuario de Amazon FSx para Windows y [Opciones de almacenamiento múltiples](#) en la Guía del usuario de Amazon FSx for Lustre.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

DeletionPolicy (Opcional, String)

Especifica si el sistema de archivos debe retenerse o eliminarse al quitarlo del clúster o al eliminar el clúster. Los valores admitidos son Delete y Retain. El valor predeterminado es Delete.

Si [DeletionPolicy](#) se establece en Delete, un sistema de archivos gestionado, con sus datos, se elimina si se elimina el clúster o si el sistema de archivos se elimina con una actualización del clúster.

Para obtener más información, consulte [Almacenamiento compartido](#).

Política de actualización: esta configuración se puede cambiar durante una actualización.

Note

DeletionPolicy admite a partir de la AWS ParallelCluster versión 3.3.0.

DataRepositoryAssociations (Opcional, String)

Lista de DRAs (hasta 8 por sistema de archivos)

Cada asociación de repositorios de datos debe tener un directorio exclusivo del sistema de archivos de Amazon FSx y un bucket o prefijo de S3 exclusivo asociado.

No se pueden utilizar [ExportPath](#) [ImportPath](#) FsxLustreSettings al mismo tiempo que se utilizan las DRAs.

Política de actualización: esta configuración se puede cambiar durante una actualización.

Name (Requerido, String)

El nombre de la DRA. Este nombre se utiliza al actualizar la configuración.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

BatchImportMetaDataOnCreate (Opcional, Boolean)

Un indicador booleano que indica si una tarea de importación de repositorios de datos que va a importar metadatos debe ejecutarse después de crear la asociación de repositorios de datos. La tarea se ejecuta si este indicador está establecido en `true`.

Valor predeterminado: `false`

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

DataRepositoryPath (Requerido, String)

La ruta al repositorio de datos de Amazon S3 que se vinculará al sistema de archivos. La ruta puede ser un bucket de S3 o un prefijo con el formato `s3://myBucket/myPrefix/`. Esta ruta especifica la parte del repositorio de datos de S3 desde donde se importarán o exportarán los archivos.

No se puede superponer con otras DRA

Patrón: `^[^\u0000\u0085\u2028\u2029\r\n]{3,4357}$`

Mínimo: 3

Máximo: 4357

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

FileSystemPath (Requerido, String)

Ruta del sistema de archivos de Amazon FSx para Lustre que apunta a un directorio de alto nivel (por ejemplo, `/ns1/`) o subdirectorio (por ejemplo, `/ns1/subdir/`) con el que se asignará 1 a 1 con `DataRepositoryPath`. Se requiere la barra diagonal que aparece al principio del nombre. Dos asociaciones de repositorios de datos no pueden tener rutas de sistema de archivos superpuestas. Por ejemplo, si un repositorio de datos está asociado a la ruta del sistema de archivos `/ns1/`, no se puede vincular otro repositorio de datos con la ruta del sistema de archivos `/ns1/ns2`.

Esta ruta especifica desde qué parte del sistema de archivos se exportarán o importarán los archivos. Este directorio del sistema de archivos solo se puede vincular a un bucket de Amazon S3; no se puede vincular ningún otro bucket de S3 al directorio.

No se puede superponer con otras DRA

Note

Si especifica solo una barra diagonal (/) como ruta del sistema de archivos, a este solo se puede vincular un repositorio de datos. Solo puede especificar / "» como la ruta del sistema de archivos para el primer repositorio de datos asociado a un sistema de archivos.

Patrón: `^[^\\u0000\\u0085\\u2028\\u2029\\r\\n]{1,4096}$`

Mínimo: 1

Máximo: 4096

Política de actualización: si se cambia esta configuración, no se permite la actualización.

ImportedFileChunkSize (Opcional, Integer)

Para los archivos que se importan desde un repositorio de datos, este valor determina la cantidad de fragmentos y la cantidad máxima de datos por archivo (en MiB) que se almacenan en un único disco físico. El número máximo de discos en los que un único archivo se puede fraccionar está limitado por el número total de los discos que forman la memoria caché o el sistema de archivos.

El valor predeterminado de fragmento es 1024 MiB (1 GiB) y puede llegar hasta 512 000 MiB (500 GiB). Los objetos de Amazon S3 tienen un tamaño máximo de 5 TB.

Mínimo: 1

Máximo: 4096

Política de actualización: esta configuración se puede cambiar durante una actualización.

AutoExportPolicy (Opcional, Array of strings)

La lista puede contener uno o más de los siguientes valores:

- NEW: los nuevos archivos y directorios se exportan automáticamente al repositorio de datos a medida que se añaden al sistema de archivos.
- CHANGED: los cambios en los archivos y directorios del sistema de archivos se exportan automáticamente al repositorio de datos.
- DELETED: los archivos y directorios se eliminan automáticamente del repositorio de datos cuando se eliminan del sistema de archivos.

Puede definir cualquier combinación de tipos de eventos para `AutoExportPolicy`.

Máximo: 3

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

`AutoImportPolicy` (Opcional, Array of strings)

La lista puede contener uno o más de los siguientes valores:

- **NEW:** Amazon FSx importa automáticamente metadatos de archivos agregados al bucket de S3 vinculado que no exista actualmente en el sistema de archivos FSx.
- **CHANGED:** Amazon FSx actualiza automáticamente los metadatos de los archivos e invalida el contenido de los archivos existentes en el sistema a medida que estos cambian en el repositorio de datos.
- **DELETED:** Amazon FSx elimina automáticamente los archivos del sistema de archivos a medida que los estos se eliminan del repositorio de datos.

Puede definir cualquier combinación de tipos de eventos para `AutoImportPolicy`.

Máximo: 3

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

FsxOntapSettings

Note

Debe definir `FsxOntapSettings` si se especifica `FsxOntap` para [StorageType](#).

(Opcional) La configuración de un sistema de archivos de FSx para ONTAP.

`FsxOntapSettings`:
`VolumeId`: *string*

Propiedades de `FsxOntapSettings`

`VolumeId` (Requerido, String)

Especifica el ID de volumen del FSx existente para el sistema ONTAP.

Note

- Si se utiliza un AWS Batch programador, FSx para ONTAP solo está disponible en el nodo principal.
- Si el tipo de implementación FSx para ONTAP es Multi-AZ, asegúrese de que la tabla de enrutamiento de la subred del nodo principal esté configurada correctamente.
- La compatibilidad con FSx para ONTAP se añadió en AWS ParallelCluster la versión 3.2.0.
- Si se usa un sistema de archivos ya existente, debe asociarse a un grupo de seguridad que permita el tráfico TCP de entrada y salida desde a través del puerto .

Asegúrese de que se permite el tráfico entre el clúster y el sistema de archivos mediante una de las siguientes acciones:

- Configure los grupos de seguridad del sistema de archivos para permitir el tráfico hacia y desde el CIDR o la lista de prefijos de las subredes del clúster.

Note

AWS ParallelCluster valida que los puertos estén abiertos y que el CIDR o la lista de prefijos estén configurados. AWS ParallelCluster no valida el contenido de la lista de bloques o prefijos del CIDR.

- Configure grupos de seguridad personalizados para los nodos del clúster mediante [SlurmQueues/Networking/SecurityGroups](#) y [HeadNode/Networking/SecurityGroups](#). Los grupos de seguridad personalizados deben configurarse para permitir el tráfico entre el clúster y el sistema de archivos.

Note

Si todos los nodos del clúster utilizan grupos de seguridad personalizados, AWS ParallelCluster solo valida que los puertos estén abiertos. AWS ParallelCluster no valida que el origen y el destino estén configurados correctamente.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

FsxOpenZfsSettings

Note

Debe definir FsxOpenZfsSettings si se especifica FsxOpenZfs para [StorageType](#).

(Opcional) La configuración de un sistema de archivos FSx para OpenZFS.

[FsxOpenZfsSettings](#):

[VolumeId](#): *string*

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Propiedades de FsxOpenZfsSettings

VolumeId (Requerido, String)

Especifica el ID de volumen del sistema FSx existente para OpenZFS.

Note

- Si se utiliza un AWS Batch programador, FSx para OpenZFS solo está disponible en el nodo principal.
- La compatibilidad con FSx para OpenZFS se agregó en la versión 3.2.0. AWS ParallelCluster
- Si se usa un sistema de archivos ya existente, debe asociarse a un grupo de seguridad que permita el tráfico TCP de entrada y salida desde a través del puerto .

Asegúrese de que se permite el tráfico entre el clúster y el sistema de archivos de la siguiente manera:

- Configure los grupos de seguridad del sistema de archivos para permitir el tráfico hacia y desde el CIDR o la lista de prefijos de las subredes del clúster.

Note

AWS ParallelCluster valida que los puertos estén abiertos y que el CIDR o la lista de prefijos estén configurados. AWS ParallelCluster no valida el contenido de la lista de bloques o prefijos del CIDR.

- Configure grupos de seguridad personalizados para los nodos del clúster mediante [SlurmQueues/Networking/SecurityGroups](#) y [HeadNode/Networking/SecurityGroups](#). Los grupos de seguridad personalizados deben configurarse para permitir el tráfico entre el clúster y el sistema de archivos.

Note

Si todos los nodos del clúster utilizan grupos de seguridad personalizados, AWS ParallelCluster solo valida que los puertos estén abiertos. AWS ParallelCluster no valida que el origen y el destino estén configurados correctamente.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

FileCacheSettings

Note

Debe definir FileCacheSettings si se especifica FileCache para [StorageType](#).

(Opcional) La configuración de una caché de archivos.

```
FileCacheSettings:  
FileCacheId: string
```

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Propiedades de FileCacheSettings

FileCacheId (Requerido, String)

Especifica el ID de caché de archivos de una caché de archivos existente.

Note

- La caché de archivos no admite AWS Batch programadores.
- Support for File Cache se añade en la AWS ParallelCluster versión 3.7.0.
- Si se usa un sistema de archivos ya existente, debe asociarse a un grupo de seguridad que permita el tráfico TCP de entrada y salida desde a través del puerto .

Asegúrese de que se permite el tráfico entre el clúster y el sistema de archivos de la siguiente manera:

- Configure los grupos de seguridad de la caché de archivos para permitir el tráfico hacia y desde el CIDR o la lista de prefijos de las subredes del clúster.

Note

AWS ParallelCluster valida que los puertos estén abiertos y que el CIDR o la lista de prefijos estén configurados. AWS ParallelCluster no valida el contenido de la lista de bloques o prefijos del CIDR.

- Configure grupos de seguridad personalizados para los nodos del clúster mediante [SlurmQueues/Networking/SecurityGroups](#) y [HeadNode/Networking/SecurityGroups](#). Los grupos de seguridad personalizados deben configurarse para permitir el tráfico entre el clúster y el sistema de archivos.

Note

Si todos los nodos del clúster utilizan grupos de seguridad personalizados, AWS ParallelCluster solo valida que los puertos estén abiertos. AWS ParallelCluster no valida que el origen y el destino estén configurados correctamente.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Sección de **Iam**

(Opcional) Especifica las propiedades de IAM del clúster.

Iam:**Roles:****LambdaFunctionsRole:** *string***PermissionsBoundary:** *string***ResourcePrefix:** *string*

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

Propiedades de Iam**PermissionsBoundary** (Opcional, String)

El ARN de la política de IAM que se utilizará como límite de permisos que se aplica a todos los roles de IAM creados por AWS ParallelCluster. Para obtener más información, consulte [Límites de permisos para las entidades de IAM](#) en la Guía del usuario de IAM. El formato es `arn:${Partition}:iam::${Account}:policy/${PolicyName}`.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

Roles (opcional)

Especifica la configuración de los roles de IAM utilizados por el clúster.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

LambdaFunctionsRole (Opcional, String)

El ARN del rol de IAM que se va a utilizar. AWS Lambda Esto anula la función predeterminada asociada a todas las funciones de Lambda que AWS CloudFormation respaldan los recursos personalizados. Lambda debe configurarse como entidad principal a la que se le permite asumir el rol. Esto no anulará la función de las funciones Lambda utilizadas para. AWS Batch El formato es `arn:${Partition}:iam::${Account}:role/${RoleName}`.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

ResourcePrefix (opcional)

Especifica una ruta o un prefijo de nombre para los recursos de IAM creados por. AWS ParallelCluster

El prefijo del recurso debe seguir las [reglas de nomenclatura especificadas](#) por IAM:

- Cada nombre puede contener hasta 128 caracteres.
- Un nombre solo puede ser una cadena sin caracteres de barra diagonal (/).

- El nombre puede tener hasta 63 caracteres.
- Una ruta debe empezar y terminar con una barra inclinada (/). Puede contener varias barras diagonales (/) entre la barra diagonal inicial y la barra final (/).
- Puede combinar la ruta y el nombre /path/name.

Especifique un nombre.

```
Iam:  
  ResourcePrefix: my-prefix
```

Especifique una ruta.

```
Iam:  
  ResourcePrefix: /org/dept/team/project/user/
```

Especifique una ruta y un nombre.

```
Iam:  
  ResourcePrefix: /org/dept/team/project/user/my-prefix
```

Si especifica ambos, devuelve un error .

```
Iam:  
  ResourcePrefix: /my-prefix
```

Se devuelve un error de configuración. Una ruta debe tener dos /. Un prefijo por sí solo no puede tener /.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Sección de **LoginNodes**

Note

Support for LoginNodes se añade en la AWS ParallelCluster versión 3.7.0.

(Opcional) Especifica la configuración del grupo de nodos de inicio de sesión.

LoginNodes:**Pools:**

- **Name:** *string*
- Count:** *integer*
- InstanceType:** *string*
- GracetimePeriod:** *integer*
- Image:**
 - CustomAmi:** *string*
- Ssh:**
 - KeyName:** *string*
- Networking:**
 - SubnetIds:**
 - *string*
 - SecurityGroups:**
 - *string*
 - AdditionalSecurityGroups:**
 - *string*
- Iam:**
 - InstanceRole:** *string*
 - InstanceProfile:** *string*
 - AdditionalIamPolicies:**
 - **Policy:** *string*

Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.

Propiedades de LoginNodes**Propiedades de Pools**

Define grupos de nodos de inicio de sesión que tienen la misma configuración de recursos. Solo se puede especificar un único campo secundario.

Pools:

- **Name:** *string*
- Count:** *integer*
- InstanceType:** *string*
- GracetimePeriod:** *integer*
- Image:**
 - CustomAmi:** *string*
- Ssh:**
 - KeyName:** *string*
- Networking:**

SubnetIds:

- *string*

SecurityGroups:

- *string*

AdditionalSecurityGroups:

- *string*

Iam:

InstanceRole: *string*

InstanceProfile: *string*

AdditionalIamPolicies:

- **Policy:** *string*

Name (Requerido String)

Especifica el nombre del paso. Se usa para etiquetar los recursos de LoginNodes.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Count (Requerido Integer)

Especifica el número de nodos de inicio de sesión que se deben mantener activos.

Política de actualización: esta configuración se puede cambiar durante una actualización.

InstanceType (Requerido String)

Especifica el tipo de instancia de Amazon EC2 que se utiliza para el nodo de inicio de sesión. La arquitectura del tipo de instancia debe ser la misma que la arquitectura utilizada para la configuración de InstanceType de Slurm.

Política de actualización: esta configuración se puede cambiar si se detiene el grupo de nodos de inicio de sesión.

GracetimePeriod (Opcional Integer)

Especifica el tiempo mínimo en minutos que transcurre entre la notificación al usuario que ha iniciado sesión de que se va a dar de baja un nodo de inicio de sesión y la detención real. Los valores válidos de GracetimePeriod oscilan entre 3 y 120 minutos. El valor predeterminado es de 60 minutos.

Note

El evento desencadenante implica interacciones entre varios AWS servicios. A veces, la latencia de la red y la propagación de la información pueden tardar algún tiempo, por lo

que el período de gracia puede tardar más de lo esperado debido a los retrasos internos en AWS los servicios.

Política de actualización: esta configuración se puede cambiar durante una actualización.

Image (opcional)

Define la configuración de imagen para los nodos de inicio de sesión.

```
Image:  
CustomAmi: String
```

CustomAmi (Opcional String)

Especifica la AMI personalizada que se utiliza para aprovisionar los nodos de inicio de sesión. Si no se especifica, el valor predeterminado es el especificado en [Sección de HeadNode](#)

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Ssh (opcional)

Define la ssh configuración de los nodos de inicio de sesión.

```
Ssh:  
KeyName: string
```

KeyName (Opcional String)

Especifica la ssh clave utilizada para iniciar sesión en los nodos de inicio de sesión. Si no se especifica, el valor predeterminado es el especificado en [Sección de HeadNode](#)

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Networking(Obligatorio)

```
Networking:  
SubnetIds:  
- string  
SecurityGroups:  
- string  
AdditionalSecurityGroups:
```

- *string*

SubnetIds (Requerido [String])

El ID de la subred existente en la que aprovisiona el grupo de nodos de inicio de sesión. Solo puede definir una subred.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

SecurityGroups (Opcional [String])

Una lista de grupos de seguridad que se utilizarán para el grupo de nodos de inicio de sesión. Si no se especifica ningún grupo de seguridad, AWS ParallelCluster crea grupos de seguridad para usted.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

AdditionalSecurityGroups (Opcional [String])

Una lista de grupos de seguridad adicionales que se utilizarán para el grupo de nodos de inicio de sesión.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Iam (opcional)

Especifica un rol de instancia o un perfil de instancia que se usará en los nodos de inicio de sesión para anular el rol de instancia o el perfil de instancia predeterminados del clúster.

Iam:

InstanceRole: *string*

InstanceProfile: *string*

AdditionalIamPolicies:

- Policy: *string*

InstanceProfile (Opcional String)

Especifica un perfil de instancia para anular el perfil de instancia del nodo de inicio de sesión predeterminado. No puede especificar InstanceProfile ni InstanceRole. El formato es `arn:Partition:iam::Account:instance-profile/InstanceProfileName`. Si se especifica, no se puede especificar la configuración de AdditionalIamPolicies y InstanceRole.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

InstanceRole (Opcional String)

Especifica un rol de instancia para anular el rol de instancia predeterminado del nodo de inicio de sesión. No puede especificar InstanceProfile ni InstanceRole. El formato es `arn:Partition:iam::Account:role/RoleName`. Si se especifica, no se puede especificar la configuración de AdditionalIamPolicies y S3Access. Si se especifica, no se puede especificar la configuración de AdditionalIamPolicies y InstanceProfile.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

AdditionalIamPolicies (opcional)

[AdditionalIamPolicies:](#)

- [Policy:](#) *string*

Nombre de recurso de Amazon (ARN) de la política de IAM que se va a consultar.

Especifica una lista separada de nombres de recursos de Amazon (ARN) de políticas de IAM para Amazon EC2. Esta lista se adjunta a la función raíz utilizada para el nodo de inicio de sesión, además de los permisos que requiere AWS ParallelCluster.

El nombre de una política de IAM y su ARN son diferentes. Los nombres no se pueden usar.

Si se especifica, no se puede especificar la configuración de InstanceRole y InstanceProfile. Se recomienda utilizarlos AdditionalIamPolicies porque AdditionalIamPolicies se añaden a los AWS ParallelCluster permisos necesarios y InstanceRole deben incluir todos los permisos necesarios. Los permisos necesarios a menudo cambian de versión a versión a medida que se añaden características.

No hay valor predeterminado.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Policy (Requerido [String])

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Sección de **Monitoring**

(Opcional) Especifica la configuración de supervisión del clúster.

Monitoring:

Logs:**CloudWatch:****Enabled:** *boolean***RetentionInDays:** *integer***DeletionPolicy:** *string***Rotation:****Enabled:** *boolean***Dashboards:****CloudWatch:****Enabled:** *boolean***DetailedMonitoring:** *boolean***Alarms:****Enabled:** *boolean*

Política de actualización: esta configuración no se analiza durante una actualización.

Propiedades de **Monitoring**

Logs (opcional)

La configuración de registro del clúster.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

CloudWatch (opcional)

La configuración de CloudWatch registros del clúster.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Enabled (Requerido, Boolean)

Si `true`, los registros del clúster se transmiten a CloudWatch Logs. El valor predeterminado es `true`.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

RetentionInDays (Opcional, Integer)

El número de días que se conservan los eventos del registro en los CloudWatch registros. El valor predeterminado es 180. Los valores admitidos son 0, 1, 3, 5, 7, 14, 30, 60, 90, 120, 150, 180, 365, 400, 545, 731, 1827 y 3653. Un valor de 0 utilizará la configuración de retención de CloudWatch registros predeterminada, es decir, no caducará nunca.

Política de actualización: esta configuración se puede cambiar durante una actualización.

DeletionPolicy (Opcional, String)

Indica si se deben eliminar los eventos de registro en CloudWatch los registros cuando se elimina el clúster. Los valores posibles son `Delete` y `Retain`. El valor predeterminado es `Retain`.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

Rotation (opcional)

La configuración de rotación de registros del clúster.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Enabled (Requerido, Boolean)

La rotación de los registros está habilitada. El valor predeterminado es `true`. Cuando un archivo de registro AWS ParallelCluster configurado alcanza un tamaño determinado, se rota y se mantiene una única copia de seguridad. Para obtener más información, consulte [Rotación de registros configurada de AWS ParallelCluster](#).

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Dashboards (opcional)

La configuración del panel de control del clúster.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

CloudWatch (opcional)

La configuración del CloudWatch panel de control del clúster.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

Enabled (Requerido, Boolean)

Si `true`, el CloudWatch panel de control está activado. El valor predeterminado es `true`.

[Política de actualización: esta configuración se puede cambiar durante una actualización.](#)

DetailedMonitoring (Opcional, Boolean)

Si se establece en `true`, se habilita la supervisión detallada para las instancias EC2 de la flota de computación. Cuando está habilitada, la consola Amazon EC2 muestra gráficos para monitorear las instancias en intervalos de 1 minuto. Cuando se habilita esta característica, hay costos adicionales. El valor predeterminado es `false`.

Para obtener más información, consulte [Habilitar o desactivar el monitoreo detallado para las instancias](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.

Note

DetailedMonitoring se añade a partir de la AWS ParallelCluster versión 3.6.0.

Alarms (opcional)

CloudWatch Alarms para el clúster.

Política de actualización: esta configuración se puede cambiar durante una actualización.

Enabled (opcional)

Si true, se crearán las CloudWatch alarmas para el clúster. El valor predeterminado es true.

Política de actualización: esta configuración se puede cambiar durante una actualización.

Note

A partir de la AWS ParallelCluster versión 3.8.0, se crean las siguientes alarmas para el nodo principal: EC2 Health Check, uso de CPU/memoria/disco y una alarma compuesta que incluye todas las demás.

Sección de Tags

(Opcional), la matriz define las etiquetas que utilizan AWS CloudFormation y se propagan a todos los recursos del clúster. Para obtener más información, consulte [AWS CloudFormation Etiqueta de recurso](#) en la Guía del usuario de AWS CloudFormation .

Tags:

- **Key:** *string*
- Value:** *string*

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Propiedades de **Tags**

Key (Requerido, String)

Define el nombre del clúster.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Value (Requerido, String)

Define el valor de etiqueta.

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Sección de **AdditionalPackages**

(Opcional) Se utiliza para identificar los paquetes adicionales que se van a instalar.

[AdditionalPackages:](#)

[IntelSoftware:](#)

[IntelHpcPlatform:](#) *boolean*

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

IntelSoftware

(Opcional) Define la configuración de las soluciones seleccionadas de Intel.

[IntelSoftware:](#)

[IntelHpcPlatform:](#) *boolean*

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Propiedades de **IntelSoftware**

IntelHpcPlatform (Opcional, Boolean)

Si existe, indica que se acepta el trueAcuerdo de licencia de usuario final para Intel Parallel Studio. Esto hace que Intel Parallel Studio se instale en el nodo maestro y se comparta con los nodos de computación. Esto añade varios minutos al tiempo que tarda el nodo principal en arrancar. El valor de IntelHpcPlatform solo se admite en 7 ().

[Política de actualización: si se cambia esta configuración, no se permite la actualización.](#)

Sección de **DirectoryService**

Note

El soporte para `DirectoryService` se agregó en la AWS ParallelCluster versión 3.1.1.

(Opcional) La configuración del servicio de directorio para un clúster que admite el acceso de varios usuarios.

AWS ParallelCluster administra los permisos que permiten el acceso de varios usuarios a los clústeres con un Active Directory (AD) mediante el Protocolo ligero de acceso a directorios (LDAP) compatible con el [Demonio de Servicios de Seguridad del Sistema \(SSSD\)](#). Para obtener más información, consulte [What is AWS Directory Service? ¿Qué es AWS Directory Service?](#) en la Guía de administración de AWS Directory Service .

Le recomendamos que utilice LDAP en lugar de TLS/SSL (abreviado LDAPS) para garantizar que cualquier información potencialmente confidencial se transmita a través de canales cifrados.

DirectoryService:

```
DomainName: string  
DomainAddr: string  
PasswordSecretArn: string  
DomainReadOnlyUser: string  
LdapTlsCaCert: string  
LdapTlsReqCert: string  
LdapAccessFilter: string  
GenerateSshKeysForUsers: boolean  
AdditionalSssdConfigs: dict
```

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

Propiedades de **DirectoryService**

Note

Si planea utilizarlos AWS ParallelCluster en una sola subred sin acceso a Internet, consulte los requisitos adicionales. [AWS ParallelCluster en una subred individual sin acceso a Internet](#)

DomainName (Requerido, String)

El dominio de Active Directory (AD) que utiliza para la información de identidad.

DomainName acepta los formatos de nombre de dominio completo (FQDN) y nombre de dominio completo (DN).

- Ejemplo de FQDN: corp.*example*.com
- Ejemplo de DN de LDAP: DC=*corp*,DC=*example*,DC=*com*

Esta propiedad corresponde al parámetro sssd-ldap que se llama. ldap_search_base

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

DomainAddr (Requerido, String)

El URI o los URI que apuntan al controlador de dominio de AD que se utiliza como servidor LDAP. El URI corresponde al parámetro SSSD-LDAP al que se llama. ldap_uri El valor puede ser una cadena de URI separada por comas. Para usar LDAP, debe agregarlo ldap:// al principio de cada URI.

Valores de ejemplo:

```
ldap://192.0.2.0,ldap://203.0.113.0          # LDAP
ldaps://192.0.2.0,ldaps://203.0.113.0     # LDAPS without support for certificate
verification
ldaps://abcdef01234567890.corp.example.com # LDAPS with support for certificate
verification
192.0.2.0,203.0.113.0                     # AWS ParallelCluster uses LDAPS by
default
```

Si usa LDAPS con verificación de certificados, los URI deben ser nombres de host.

Si utiliza LDAPS sin verificación de certificado o LDAP, los URI pueden ser nombres de host o direcciones IP.

Utilice LDAP sobre TLS/SSL (LDAPS) para evitar la transmisión de contraseñas y otra información confidencial a través de canales no cifrados. Si AWS ParallelCluster no encuentra un protocolo, lo añade ldaps:// al principio de cada URI o nombre de host.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

PasswordSecretArn (Requerido, String)

El nombre de recurso de Amazon (ARN) del AWS Secrets Manager secreto que contiene la contraseña en texto DomainReadOnlyUser simple. El contenido del secreto corresponde al parámetro SSSD-LDAP al que se llama. ldap_default_authtok

Note

Al crear un secreto con la AWS Secrets Manager consola, asegúrese de seleccionar «Otro tipo de secreto», seleccionar texto sin formato e incluir solo el texto de la contraseña en el secreto.

Para obtener más información sobre cómo AWS Secrets Manager crear un secreto, consulta [Crear un AWS Secrets Manager secreto](#)

El cliente LDAP usa la contraseña para autenticarse en el dominio AD cuando solicita información de identidad. DomainReadOnlyUser

Si el usuario tiene el permiso para [DescribeSecret](#), se valida PasswordSecretArn. PasswordSecretArn es válido si el secreto especificado existe. Si la política de IAM del usuario no la incluye DescribeSecret, PasswordSecretArn no está validada y aparece un mensaje de advertencia. Para obtener más información, consulte [Política de usuario básica pcluster de AWS ParallelCluster](#).

Cuando el valor del secreto cambia, el clúster no se actualiza automáticamente. Para actualizar el clúster para el nuevo valor secreto, debe detener la flota de computación con el comando [the section called “pcluster update-compute-fleet”](#) y, luego, ejecutar el siguiente comando desde dentro del nodo principal.

```
$ sudo /opt/parallelcluster/scripts/directory_service/  
update_directory_service_password.sh
```

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

DomainReadOnlyUser (Requerido, String)

La identidad que se usa para consultar la información de identidad en el dominio de AD al autenticar los inicios de sesión de los usuarios del clúster. Corresponde al parámetro SSSD-

LDAP al que se llama. `ldap_default_bind_dn` Use su información de identidad de AD para este valor.

Especifique la identidad en el formulario requerido por el cliente LDAP específico que se encuentra en el nodo:

- Microsoft AD:

```
cn=ReadOnlyUser,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
```

- SimpleAD:

```
cn=ReadOnlyUser,cn=Users,dc=corp,dc=example,dc=com
```

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

LdapTlsCaCert (Opcional, String)

La ruta absoluta a un paquete de certificados que contiene los certificados de cada entidad emisora de certificados de la cadena de certificación que emitió un certificado para los controladores de dominio. Corresponde al parámetro SSSD-LDAP que se llama. `ldap_tls_cacert`

Un paquete de certificados es un archivo compuesto por la concatenación de distintos certificados en formato PEM, también conocido como formato DER Base64 en Windows. Se utiliza para comprobar la identidad del controlador de dominio de AD que actúa como servidor LDAP.

AWS ParallelCluster no es responsable de la colocación inicial de los certificados en los nodos. Como administrador del clúster, puede configurar el certificado en el nodo principal manualmente después de crear el clúster o puede utilizar un [script de arranque](#). También puede utilizar una Imagen de máquina de Amazon (AMI) que incluya el certificado configurado en el nodo principal.

[Simple AD](#) no admite LDAPS. Para obtener información sobre cómo integrar un directorio AD simple AWS ParallelCluster, consulte [Cómo configurar un punto final LDAPS para AD simple](#) en el blog de AWS seguridad.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

LdapTlsReqCert (Opcional, String)

Especifica qué comprobaciones se deben realizar en los certificados de servidor en una sesión de TLS. Corresponde al parámetro SSSD-LDAP que se llama. `ldap_tls_reqcert`

Valores válidos: `never`, `allow`, `try`, `demand` y `hard`.

`neverallow`, y `try` permiten que las conexiones continúen aunque se detecten problemas con los certificados.

`demandy hard` permita que la comunicación continúe si no se detecta ningún problema con los certificados.

Si el administrador del clúster utiliza un valor que no requiere que la validación del certificado se realice correctamente, se le devuelve un mensaje de advertencia. Por motivos de seguridad, recomendamos que no inhabilite la verificación de certificados.

El valor predeterminado es `hard`.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

LdapAccessFilter (Opcional, String)

Especifica un filtro para limitar el acceso al directorio a un subconjunto de usuarios. Esta propiedad corresponde al parámetro SSSD-LDAP que se llama. `ldap_access_filter` Puede usarla para limitar las consultas a un AD que admita un gran número de usuarios.

Este filtro puede bloquear el acceso de los usuarios al clúster. Sin embargo, no afecta a la capacidad de detección de los usuarios bloqueados.

Si se establece esta propiedad, el parámetro SSSD `access_provider` se establece `Ldap` internamente mediante la configuración [DirectoryService/ AWS ParallelCluster AdditionalSssdConfigs](#) y no se debe modificar mediante ella.

Si se omite esta propiedad y el acceso personalizado de los usuarios no se especifica en [DirectoryService/AdditionalSssdConfigs](#), todos los usuarios del directorio pueden acceder al clúster.

Ejemplos:

```
"!(cn=SomeUser*)" # denies access to every user with alias starting with "SomeUser"
```

```
"(cn=SomeUser*)" # allows access to every user with alias starting with "SomeUser"  
"memberOf=cn=TeamOne,ou=Users,ou=CORP,dc=corp,dc=example,dc=com" # allows access  
only to users in group "TeamOne".
```

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

GenerateSshKeysForUsers (Opcional, Boolean)

Define si se AWS ParallelCluster genera una clave SSH para los usuarios del clúster inmediatamente después de su autenticación inicial en el nodo principal.

Si se establece en `true`, se genera una clave SSH y se guarda `USER_HOME_DIRECTORY/.ssh/id_rsa`, si no existe, para cada usuario tras su primera autenticación en el nodo principal.

En el caso de un usuario que aún no se ha autenticado en el nodo principal, la primera autenticación se puede realizar en los siguientes casos:

- El usuario inicia sesión en el nodo principal por primera vez con su propia contraseña.
- En el nodo principal, un `sudoer` cambia al usuario por primera vez: `su USERNAME`
- En el nodo principal, un `sudoer` ejecuta un comando como usuario por primera vez: `su -u USERNAME COMMAND`

Los usuarios pueden usar la clave SSH para iniciar sesión posteriormente en el nodo principal del clúster y en los nodos de cómputo. Con AWS ParallelCluster, los inicios de sesión con contraseña en los nodos de cómputo del clúster están deshabilitados por diseño. Si un usuario no ha iniciado sesión en el nodo principal, las claves SSH no se generan y el usuario no podrá iniciar sesión en los nodos de cálculo.

El valor predeterminado es `true`.

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

AdditionalSssdConfigs (Opcional, Dict)

Un diccionario de pares clave-valor que contiene parámetros y valores de SSSD para escribirlos en el archivo de configuración de SSSD en las instancias del clúster. Para obtener una descripción completa del archivo de configuración de SSSD, consulte las páginas del manual de la instancia para SSSD y los archivos de configuración relacionados.

Los parámetros y valores del SSSD deben ser compatibles con la configuración AWS ParallelCluster del SSSD, tal y como se describe en la siguiente lista.

- `id_provider` está configurado `ldap` internamente por AWS ParallelCluster y no debe modificarse.
- `access_provider` establece `ldap` internamente AWS ParallelCluster cuando [LdapAccessFilter](#) se especifica [DirectoryService/](#), y esta configuración no debe modificarse.

Si [LdapAccessFilter](#) se omite [DirectoryService/](#), también se omite su `access_provider` especificación. Por ejemplo, si se establece `access_provider simple` en [AdditionalSssdConfigs](#), no se [LdapAccessFilter](#) debe especificar [DirectoryService/](#).

Los siguientes fragmentos de configuración son ejemplos de configuraciones válidas para `AdditionalSssdConfigs`

En este ejemplo, se habilita el nivel de depuración de los registros SSSD, se restringe la base de búsqueda a una unidad organizativa específica y se deshabilita el almacenamiento en caché de las credenciales.

```
DirectoryService:
  ...
  AdditionalSssdConfigs:
    debug_level: "0xFFF0"
    ldap_search_base: OU=Users,OU=CORP,DC=corp,DC=example,DC=com
    cache_credentials: False
```

En este ejemplo se especifica la configuración de un `access_provider simple` SSSD. Los usuarios del `EngineeringTeam` tienen acceso al directorio. [DirectoryService/](#)no [LdapAccessFilter](#) debe configurarse en este caso.

```
DirectoryService:
  ...
  AdditionalSssdConfigs:
    access_provider: simple
    simple_allow_groups: EngineeringTeam
```

[Política de actualización: la flota de computación debe estar detenida para poder cambiar esta configuración y proceder a la actualización.](#)

Sección de **DeploymentSettings**

Note

DeploymentSettings se añade a partir de la AWS ParallelCluster versión 3.4.0.

(Opcional) Especifica la configuración de los ajustes de implementación.

```
DeploymentSettings:  
  LambdaFunctionsVpcConfig:  
    SecurityGroupIds  
      - string  
    SubnetIds  
      - string  
  DisableSudoAccessForDefaultUser: Boolean  
  DefaultUserHome: string # 'Shared' or 'Local'
```

Propiedades de **DeploymentSettings**

LambdaFunctionsVpcConfig

(Opcional) Especifica las configuraciones de VPC de AWS Lambda las funciones. Para obtener más información, consulte [Configuración de la VPC de AWS Lambda en AWS ParallelCluster](#).

```
LambdaFunctionsVpcConfig:  
  SecurityGroupIds  
    - string  
  SubnetIds  
    - string
```

LambdaFunctionsVpcConfig properties

SecurityGroupIds (Requerido, [String])

La lista de los ID de los grupos de seguridad de Amazon VPC que se adjuntan a las funciones de Lambda.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

SubnetIds (Requerido, [String])

La lista de identificadores de subred que se adjuntan a las funciones de Lambda.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Note

Las subredes y los grupos de seguridad deben estar en la misma VPC.

DisableSudoAccessForDefaultUser propiedad

Note

Esta opción de configuración solo es compatible con los clústeres de Slurm.

(Opcional) Si `True`, se deshabilitarán los privilegios de sudo del usuario predeterminado. Esto se aplica a todos los nodos del clúster.

```
# Main DeploymentSettings section in config yaml(applyes to HN, CF and LN)
DeploymentSettings:
  DisableSudoAccessForDefaultUser: True
```

Para actualizar el valor de `DisableSudoAccessForDefaultUser`, debe detener la flota de procesamiento y todos los nodos de inicio de sesión.

Política de actualización: la flota de cómputo y los nodos de inicio de sesión deben estar detenidos para poder cambiar esta configuración y poder actualizarse.

DefaultUserVivienda, propiedad

Si se establece en `Shared`, el clúster utilizará la configuración predeterminada y compartirá el directorio de usuarios predeterminado en todo el clúster de una forma u otra `/home/<default user>`.

Si se configura en `Local`, el nodo principal, los nodos de inicio de sesión y los nodos de cómputo tendrán cada uno un directorio de usuarios local predeterminado independiente almacenado `local/home/<default user>`.

Cree archivos de configuración de imágenes

AWS ParallelCluster la versión 3 usa archivos YAML 1.1 para compilar los parámetros de configuración de la imagen. Confirme que la indentación es correcta para reducir los errores de configuración. Para obtener más información, consulte la especificación de YAML 1.1 en <https://yaml.org/spec/1.1/>

Estos archivos de configuración se utilizan para definir cómo se crean las AWS ParallelCluster AMI personalizadas mediante EC2 Image Builder. Los procesos de creación de AMI personalizados se activan mediante el `pcluster build-image` comando. Para ver algunos ejemplos de archivos de configuración, consulte https://github.com/aws/aws-parallelcluster/tree/release-3.0/cli/tests/pcluster/schemas/test_imagebuilder_schema/test_imagebuilder_schema.

Temas

- [Cree las propiedades del archivo de configuración de imágenes](#)
- [Sección de Build](#)
- [Sección de Image](#)
- [Sección de DeploymentSettings](#)

Cree las propiedades del archivo de configuración de imágenes

Region (Opcional, String)

Especifica el Región de AWS para la `build-image` operación. Por ejemplo, `us-east-2`.

Sección de **Build**

(Obligatorio) Especifica la configuración en la que se generará la imagen.

```
Build:
  Imds:
    ImdsSupport: string
  InstanceType: string
  SubnetId: string
  ParentImage: string
  Iam:
    InstanceRole: string
    InstanceProfile: string
```

```
CleanupLambdaRole: string
AdditionalIamPolicies:
  - Policy: string
PermissionsBoundary: string
Components:
  - Type: string
    Value: string
Tags:
  - Key: string
    Value: string
SecurityGroupIds:
  - string
UpdateOsPackages:
  Enabled: boolean
```

Propiedades de **Build**

InstanceType (Requerido, String)

Especifica el tipo de instancia de la instancia utilizada para crear la imagen.

SubnetId (Opcional, String)

Especifica el ID de una subred existente en la que se va a aprovisionar la instancia para generar la imagen. La subred proporcionada requiere acceso a Internet.

Warning

`pcluster build-image` utiliza la VPC predeterminada. Si se ha eliminado la VPC predeterminada, quizás mediante AWS Control Tower una zona de AWS aterrizaje, se debe especificar el ID de subred.

ParentImage (Requerido, String)

Especifica la imagen base. La imagen principal puede ser una AMI que no sea una AWS ParallelCluster AMI o una AWS ParallelCluster AMI oficial de la misma versión. No puedes usar una AMI AWS ParallelCluster oficial o personalizada de una versión diferente de AWS ParallelCluster. El formato debe ser el ARN de una imagen `arn:Partition:imagebuilder:Region:Account:image/ImageName/ImageVersion` o un ID de AMI. `ami-12345678`

SecurityGroupIds (Opcional, [String])

Especifica la lista de los ID de los grupos de seguridad de la imagen.

Imds

Propiedades de Imds

(Opcional) Especifica la configuración del servicio de metadatos de instancias de ImageBuilder compilación y prueba (IMDS) de EC2.

Imds:

ImdsSupport: *string*

ImdsSupport (Opcional, String)

Especifica qué versiones de IMDS se admiten en las instancias de ImageBuilder compilación y prueba de EC2. Los valores admitidos son `v2.0` y `v1.0`. El valor predeterminado es `v2.0`.

Si `ImdsSupport` se establece en `v1.0`, se admiten tanto `IMDSv1` como `IMDSv2`.

Si `ImdsSupport` se establece en `v2.0`, solo se admite `IMDSv2`.

Para obtener más información, consulte [Utilizar IMDSv2](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Note

A partir de AWS ParallelCluster la versión 3.7.0, el `ImdsSupport` valor predeterminado es `v2.0`. Le recomendamos que configure `ImdsSupport` en `v2.0` y sustituya `IMDSv1` por `IMDSv2` en sus llamadas a acciones personalizadas.

El soporte para [Imds/ImdsSupport](#) se añade con la AWS ParallelCluster versión 3.3.0.

Iam

Propiedades de Iam

(Opcional) Especifica los recursos de IAM para la creación de la imagen.

Iam:

```
InstanceRole: string  
InstanceProfile: string  
CleanupLambdaRole: string  
AdditionalIamPolicies:  
  - Policy: string  
PermissionsBoundary: string
```

InstanceProfile (Opcional, String)

Especifica un perfil de instancia para anular el perfil de instancia predeterminado de la instancia de Generador de imágenes de EC2. InstanceProfile, InstanceRole y AdditionalIamPolicies no se pueden especificar juntos. El formato es `arn:Partition:iam::Account:instance-profile/InstanceProfileName`.

InstanceRole (Opcional, String)

Especifica un rol de instancia para anular el rol de instancia predeterminado para la instancia de Generador de imágenes de EC2. InstanceProfile, InstanceRole y AdditionalIamPolicies no se pueden especificar juntos. El formato es `arn:Partition:iam::Account:role/RoleName`.

CleanupLambdaRole (Opcional, String)

El ARN del rol de IAM que se utilizará para la AWS Lambda función que respalda el recurso AWS CloudFormation personalizado que elimina los artefactos de compilación al finalizar la compilación. Lambda debe configurarse como entidad principal a la que se le permite asumir el rol. El formato es `arn:Partition:iam::Account:role/RoleName`.

AdditionalIamPolicies (opcional)

Especifica políticas de IAM adicionales para adjuntarlas a la instancia de Generador de imágenes de EC2 utilizada para producir la AMI personalizada.

```
AdditionalIamPolicies:  
  - Policy: string
```

Policy (Opcional, [String])

Lista de políticas de IAM. El formato es `arn:Partition:iam::Account:policy/PolicyName`.

PermissionsBoundary (Opcional, String)

El ARN de la política de IAM que se utilizará como límite de permisos que se aplica a todos los roles de IAM creados por AWS ParallelCluster. Para obtener más información sobre los límites de los permisos de IAM, consulte [Límites de permisos para las entidades de IAM](#) en la Guía del usuario de IAM. El formato es `arn:Partition:iam::Account:policy/PolicyName`.

Components

Propiedades de Components

(Opcional) Especifica ImageBuilder los componentes de EC2 que se utilizarán durante el proceso de creación de la AMI, además de los que proporciona de forma predeterminada AWS ParallelCluster. Estos componentes se pueden utilizar para personalizar el proceso de creación de la AMI. Para obtener más información, consulte [AWS ParallelCluster Personalización de AMI](#).

Components:

- Type: *string*
Value: *string*

Type (Opcional, String)

Especifica el tipo de par tipo-valor del componente. El tipo puede ser `arn` o `script`

Value (Opcional, String)

Especifica el valor del par tipo-valor del componente. Si el tipo es `arn`, es el ARN de un componente de Generador de imágenes de EC2. Si el tipo es `script`, es el enlace `https` o `s3` que apunta al script que se utilizará al crear el componente Generador de imágenes de EC2.

Tags

Propiedades de Tags

(Opcional) Especifica la lista de etiquetas que se van a configurar en los recursos utilizados para crear la AMI.

Tags:

- Key: *string*
Value: *string*

Key (Opcional, String)

Define el nombre del clúster.

Value (Opcional, String)

Define el valor de etiqueta.

UpdateOsPackages

Propiedades de UpdateOsPackages

(Opcional) Especifica si el sistema operativo se actualiza antes de instalar la pila de AWS ParallelCluster software.

UpdateOsPackages:

Enabled: *boolean*

Enabled (Opcional, Boolean)

Si `true`, el sistema operativo se actualiza y se reinicia antes de instalar el AWS ParallelCluster software. El valor predeterminado es `false`.

Note

Cuando `UpdateOsPackages` está activado, se actualizan todos los paquetes de sistema operativo disponibles, incluido el núcleo. Como cliente, usted es responsable de comprobar que la actualización es compatible con las dependencias de la AMI que no se incluyen en la actualización.

Por ejemplo, supongamos que está creando una AMI para la AWS ParallelCluster versión X.0 que se incluye con la versión Y.0 del núcleo y algunos componentes de la versión Z.0. Supongamos que la actualización disponible incluye la versión Y.1 del núcleo actualizada sin actualizaciones del componente Z.0. Antes de activarlo `UpdateOsPackages`, es su responsabilidad comprobar que el componente Z.0 es compatible con el núcleo Y.1.

Sección de Image

(Opcional) Define las propiedades de la imagen para su creación.

Image:**Name:** *string***RootVolume:****Size:** *integer***Encrypted:** *boolean***KmsKeyId:** *string***Tags:**- **Key:** *string***Value:** *string*

Propiedades de **Image**

Name (Opcional, String)

Especifica el nombre de la AMI. Si no se especifica, se utiliza el nombre utilizado al llamar al [pcluster build-image](#) comando.

Tags

Propiedades de **Tags**

(Opcional) Especifica los pares clave-valor para la imagen.

Tags:- **Key:** *string***Value:** *string*

Key (Opcional, String)

Define el nombre del clúster.

Value (Opcional, String)

Define el valor de etiqueta.

RootVolume

Propiedades de **RootVolume**

(Opcional) Especifica las propiedades del volumen raíz de la imagen.

RootVolume:**Size:** *integer***Encrypted:** *boolean***KmsKeyId:** *string***Size (Opcional, Integer)**

Especifica el tamaño del volumen raíz de la imagen, en GiB. El tamaño predeterminado es el tamaño del [ParentImage](#) más de 27 GiB.

Encrypted (Opcional, Boolean)

Especifica si el volumen de está cifrado. El valor predeterminado es `false`.

KmsKeyId (Opcional, String)

Especifica el ARN de la AWS KMS clave utilizada para cifrar el volumen. El formato es el siguiente `arn:Partition:kms:Region:Account:key/KeyId`.

Sección de **DeploymentSettings**

(Opcional) Especifica la configuración de los ajustes de implementación.

DeploymentSettings:**LambdaFunctionsVpcConfig:****SecurityGroupIds**

- *string*

SubnetIds

- *string*

Propiedades de **DeploymentSettings****LambdaFunctionsVpcConfig**

(Opcional) Especifica las configuraciones de VPC de AWS Lambda las funciones. Para obtener más información, consulte [Configuración de la VPC de AWS Lambda en AWS ParallelCluster](#).

LambdaFunctionsVpcConfig:**SecurityGroupIds**

- *string*

SubnetIds

- *string*

LambdaFunctionsVpcConfig properties

SecurityGroupIds (Requerido, [String])

La lista de los ID de los grupos de seguridad de Amazon VPC que se adjuntan a las funciones de Lambda.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

SubnetIds (Requerido, [String])

La lista de identificadores de subred que se adjuntan a las funciones de Lambda.

Política de actualización: si se cambia esta configuración, no se permite la actualización.

Note

Las subredes y los grupos de seguridad deben estar en la misma VPC.

Note

DeploymentSettings se añade a partir de la AWS ParallelCluster versión 3.4.0.

Referencia de la API de AWS ParallelCluster

En esta sección se ofrecen descripciones, sintaxis y ejemplos de uso de cada una de las acciones de la API de AWS ParallelCluster.

Temas

- [buildImage](#)
- [createCluster](#)
- [deleteCluster](#)
- [deleteClusterInstances](#)
- [deleteImage](#)

- [describeCluster](#)
- [describeClusterInstances](#)
- [describeComputeFleet](#)
- [describelImage](#)
- [getClusterLogEvents](#)
- [getClusterStackEvents](#)
- [getImageLogEvents](#)
- [getImageStackEvents](#)
- [listClusters](#)
- [listClusterLogStreams](#)
- [listImageLogStreams](#)
- [listImages](#)
- [listOfficialImages](#)
- [updateCluster](#)
- [updateComputeFleet](#)

buildImage

Cree una imagen personalizada de AWS ParallelCluster en una Región de AWS.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
POST /v3/images/custom
{
```

```
"imageConfiguration": "string",
"imageId": "string",
"dryrun": boolean,
"region": "string",
"rollbackOnFailure": boolean,
"supressValidators": [ "string" ],
"validationFailureLevel": "string"
}
```

Cuerpo de la solicitud

imageConfiguration

La configuración de la imagen como documento YAML.

Tipo: String

Obligatorio: sí

imageId

El identificador de la imagen que desea crear.

Tipo: String

Obligatorio: sí

dryrun

Si se establece en `true`, solo realiza la validación de la solicitud sin crear ningún recurso. Use este parámetro para validar la configuración de la imagen. El valor predeterminado es `false`.

Tipo: booleano

Obligatorio: no

region

La Región de AWS en la que ejecuta el comando para crear la imagen.

Tipo: String

Obligatorio: no

rollbackOnFailure

Si se establece en `true`, se produce una reversión de la pila de imágenes si la imagen no se crea. El valor predeterminado es `false`.

Tipo: booleano

Obligatorio: no

suppressValidators

Identifica uno o más validadores de configuración para suprimirlos.

Tipo: lista de cadenas

Formato: (ALL | type: [A-Za-z0-9]+)

Obligatorio: no

validationFailureLevel

El nivel mínimo de validación que provoca un error en la creación de la imagen. El valor predeterminado es `ERROR`.

Tipo: String

Valores válidos: INFO | WARNING | ERROR

Obligatorio: no

Sintaxis de la respuesta

```
{
  "image": {
    "imageId": "string",
    "ec2AmiInfo": {
      "amiId": "string"
    },
    "region": "string",
    "version": "string",
    "cloudformationStackArn": "string",
    "imageBuildStatus": "BUILD_IN_PROGRESS",
    "cloudformationStackStatus": "CREATE_IN_PROGRESS"
  },
}
```

```
"validationMessages": [  
  {  
    "id": "string",  
    "type": "string",  
    "level": "INFO",  
    "message": "string"  
  }  
]  
}
```

Cuerpo de respuesta

image

imageId

El identificador de la imagen.

Tipo: String

cloudformationStackArn

El nombre de recurso de Amazon (ARN) de la pila principal de CloudFormation.

Tipo: String

cloudformationStackStatus

El estado de la pila de CloudFormation.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| ROLLBACK_IN_PROGRESS | ROLLBACK_FAILED | ROLLBACK_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_COMPLETE | UPDATE_ROLLBACK_IN_PROGRESS |
UPDATE_ROLLBACK_FAILED | UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_ROLLBACK_COMPLETE

ec2AmiInfo

ami-id

El identificador de la AMI de EC2.

Tipo: String

imageBuildStatus

El estado de creación de la imagen.

Tipo: String

Valores válidos: BUILD_IN_PROGRESS | BUILD_FAILED | BUILD_COMPLETE | DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE

region

La Región de AWS en la que se crea la imagen.

Tipo: String

version

La versión de AWS ParallelCluster que se utiliza para crear la imagen.

Tipo: String

validationMessages

Una lista de mensajes con un nivel de validación inferior a `validationFailureLevel`. La lista de mensajes se recopila durante la validación de la configuración.

id

El identificador del validador.

Tipo: String

level

El nivel de validación.

Tipo: String

Valores válidos: INFO | WARNING | ERROR

message

Un mensaje de validación.

Tipo: String

type

El tipo de validador.

Tipo: String

Ejemplo

Python

Solicitud

```
$ build_image(custom-image-id, custom-image-config.yaml)
```

Respuesta 200

```
{
  'image': {
    'cloudformation_stack_arn': 'arn:aws:cloudformation:us-
east-1:123456789012:stack/custom-image-id/711b76b0-af81-11ec-a29f-0ee549109f1f',
    'cloudformation_stack_status': 'CREATE_IN_PROGRESS',
    'image_build_status': 'BUILD_IN_PROGRESS',
    'image_id': 'custom-image-id',
    'region': 'us-east-1',
    'version': '3.2.1'
  }
}
```

createCluster

Cree un clúster administrado en una Región de AWS.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
POST /v3/clusters
{
  "clusterName": "string",
  "clusterConfiguration": "string",
  "dryrun": boolean,
  "region": "string",
  "rollbackOnFailure", boolean,
  "suppressValidators": [ "string" ],
  "validationFailureLevel": "string"
}
```

Cuerpo de la solicitud

clusterConfiguration

La configuración del clúster como documento YAML.

Tipo: String

Obligatorio: sí

clusterName

El nombre del clúster que se creará.

El nombre debe empezar por un carácter alfabético. El nombre puede tener un máximo de 60 caracteres. Si la contabilidad de Slurm está habilitada, el nombre puede tener hasta 40 caracteres.

Tipo: String

Obligatorio: sí

dryrun

Si se establece en `true`, solo realiza la validación de la solicitud sin crear ningún recurso. Use este parámetro para validar la configuración del clúster. El valor predeterminado es `false`.

Tipo: booleano

Obligatorio: no

region

La Región de AWS en la que se encuentra el clúster.

Tipo: String

Obligatorio: no

rollbackOnFailure

Si se establece en `true`, se produce una reversión de la pila del clúster si el clúster no se crea. El valor predeterminado es `true`.

Tipo: booleano

Obligatorio: no

suppressValidators

Identifica uno o más validadores de configuración para suprimirlos.

Tipo: lista de cadenas

Formato: (ALL | type: [A-Za-z0-9]+)

Obligatorio: no

validationFailureLevel

El nivel mínimo de validación que provoca un error en la creación del clúster. El valor predeterminado es `ERROR`.

Tipo: String

Valores válidos: `INFO` | `WARNING` | `ERROR`

Obligatorio: no

Sintaxis de la respuesta

```
{
  "cluster": {
    "clusterName": "string",
```

```
"region": "string",
"version": "string",
"cloudformationStackArn": "string",
"cloudformationStackStatus": "CREATE_IN_PROGRESS",
"clusterStatus": "CREATE_IN_PROGRESS",
"scheduler": {
  "type": "string",
  "metadata": {
    "name": "string",
    "version": "string"
  }
},
"validationMessages": [
  {
    "id": "string",
    "type": "string",
    "level": "INFO",
    "message": "string"
  }
]
```

Cuerpo de respuesta

clusterName

El nombre del clúster.

Tipo: String

cloudformationStackArn

El nombre de recurso de Amazon (ARN) de la pila principal de CloudFormation.

Tipo: String

cloudformationStackStatus

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| ROLLBACK_IN_PROGRESS | ROLLBACK_FAILED | ROLLBACK_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |

UPDATE_IN_PROGRESS | UPDATE_COMPLETE_CLEANUP_IN_PROGRESS |
UPDATE_COMPLETE | UPDATE_ROLLBACK_IN_PROGRESS | UPDATE_ROLLBACK_FAILED |
UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS | UPDATE_ROLLBACK_COMPLETE

clusterStatus

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE | UPDATE_FAILED

region

La Región de AWS en la que se crea el clúster.

Tipo: String

programador

metadatos

Los metadatos del programador.

nombre

El nombre del programador.

Tipo: String

version

La versión del programador.

Tipo: String

type

El tipo de programador.

Tipo: String

version

La versión de AWS ParallelCluster que se utiliza para crear el clúster.

Tipo: String

validation_messages

Una lista de mensajes con un nivel de validación inferior a `validationFailureLevel`. La lista de mensajes se recopila durante la validación de la configuración.

id

El identificador del validador.

Tipo: String

level

Tipo: String

Valores válidos: INFO | WARNING | ERROR

message

Un mensaje de validación.

Tipo: String

type

El tipo de validador.

Tipo: String

Ejemplo

Python

Solicitud

```
$ create_cluster(cluster_name_3x, cluster-config.yaml)
```

Respuesta 200

```
{
  'cluster': {
    'cloudformation_stack_arn': 'arn:aws:cloudformation:us-
east-1:123456789012:stack/cluster-3x/e0462730-50b5-11ed-99a3-0a5ddc4a34c7',
```

```
'cloudformation_stack_status': 'CREATE_IN_PROGRESS',
'cluster_name': 'cluster-3x',
'cluster_status': 'CREATE_IN_PROGRESS',
'region': 'us-east-1',
'scheduler': {
  'type': 'slurm'
},
'version': '3.2.1'
}
}
```

deleteCluster

Inicia la eliminación de un clúster.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
DELETE /v3/clusters/{clusterName}
{
  "region": "string"
}
```

Cuerpo de la solicitud

clusterName

El nombre del clúster.

Tipo: String

Obligatorio: sí

region

La Región de AWS en la que se elimina el clúster.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{
  "cluster": {
    "clusterName": "string",
    "region": "string",
    "version": "string",
    "cloudformationStackArn": "string",
    "cloudformationStackStatus": "DELETE_IN_PROGRESS",
    "clusterStatus": "DELETE_IN_PROGRESS",
    "scheduler": {
      "type": "string",
      "metadata": {
        "name": "string",
        "version": "string"
      }
    }
  }
}
```

Cuerpo de respuesta

clúster

Una lista de instancias de clúster

clusterName

El nombre de un clúster.

Tipo: String

cloudformationStackArn

El nombre de recurso de Amazon (ARN) de la pila principal de CloudFormation.

Tipo: String

cloudformationStackStatus

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| ROLLBACK_IN_PROGRESS | ROLLBACK_FAILED | ROLLBACK_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_COMPLETE | UPDATE_ROLLBACK_IN_PROGRESS |
UPDATE_ROLLBACK_FAILED | UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_ROLLBACK_COMPLETE

clusterStatus

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE | UPDATE_FAILED

region

La Región de AWS en la que se crea el clúster.

Tipo: String

programador

metadatos

Los metadatos del programador.

nombre

El nombre del programador.

Tipo: String

version

La versión del programador.

Tipo: String

type

El tipo de programador.

Tipo: String

version

La versión de AWS ParallelCluster que se utiliza para crear el clúster.

Tipo: String

Ejemplo

Python

Solicitud

```
$ delete_cluster(cluster_name_3x)
```

Respuesta 200

```
{
  'cluster': {
    'cloudformation_stack_arn': 'arn:aws:cloudformation:us-
east-1:123456789012:stack/cluster_name_3x/16b49540-ae5-11ec-8e18-0ac1d712b241',
    'cloudformation_stack_status': 'DELETE_IN_PROGRESS',
    'cluster_name': 'cluster_name_3x',
    'cluster_status': 'DELETE_IN_PROGRESS',
    'region': 'us-east-1',
    'version': '3.2.1'
  }
}
```

deleteClusterInstances

Fuerce la finalización de todos los nodos de computación del clúster. Esta acción no admite clústeres de AWS Batch.

Temas

- [Sintaxis de la solicitud](#)

- [Cuerpo de la solicitud](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
DELETE /v3/clusters/{clusterName}/instances
{
  "force": boolean,
  "region": "string"
}
```

Cuerpo de la solicitud

clusterName

El nombre del clúster.

Tipo: String

Obligatorio: sí

force

Si se establece en `true`, se fuerza la eliminación cuando no se encuentra el clúster con el nombre indicado. El valor predeterminado es `false`.

Tipo: booleano

Obligatorio: no

region

La Región de AWS en la que se encuentra el clúster.

Tipo: String

Obligatorio: no

Cuerpo de respuesta

Ninguno

Ejemplo

Python

Solicitud

```
$ delete_cluster_instances(cluster_name_3x)
```

Respuesta 200

Ninguno

deletelImage

Inicia la eliminación de la imagen personalizada de AWS ParallelCluster.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
DELETE /v3/images/custom/{imageId}
{
  "force": boolean,
  "region": "string"
}
```

Cuerpo de la solicitud

imageId

El identificador de la imagen.

Tipo: String

Obligatorio: sí

force

Si se establece en `true`, se fuerza la eliminación de la AMI. Utilice este parámetro si hay instancias que utilizan la AMI o si la AMI es compartida. El valor predeterminado es `false`.

Tipo: booleano

Obligatorio: no

region

La Región de AWS en la que se creó la imagen.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{
  "image": {
    "imageId": "string",
    "ec2AmiInfo": {
      "amiId": "string"
    },
    "region": "string",
    "version": "string",
    "cloudformationStackArn": "string",
    "imageBuildStatus": "DELETE_IN_PROGRESS",
    "cloudformationStackStatus": "DELETE_IN_PROGRESS"
  }
}
```

Cuerpo de respuesta

image

cloudformationStackArn

El nombre de recurso de Amazon (ARN) de la pila principal de CloudFormation.

Tipo: String

cloudformationStackStatus

El estado de la pila de CloudFormation.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| ROLLBACK_IN_PROGRESS | ROLLBACK_FAILED | ROLLBACK_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_COMPLETE | UPDATE_ROLLBACK_IN_PROGRESS |
UPDATE_ROLLBACK_FAILED | UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_ROLLBACK_COMPLETE

ec2AmiInfo

amild

El identificador de la AMI de EC2.

Tipo: String

imageBuildStatus

El estado de creación de la imagen.

Tipo: String

Valores válidos: BUILD_IN_PROGRESS | BUILD_FAILED | BUILD_COMPLETE |
DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE

imageId

El identificador de la imagen.

Tipo: String

region

La Región de AWS en la que se crea la imagen.

Tipo: String

version

La versión de AWS ParallelCluster que se utiliza para crear la imagen.

Tipo: String

Ejemplo

Python

Solicitud

```
$ delete_image(custom-image-id)
```

Respuesta 200

```
{
  'image': {
    'image_build_status': 'DELETE_IN_PROGRESS',
    'image_id': 'custom-image-id',
    'region': 'us-east-1',
    'version': '3.2.1'
  }
}
```

describeCluster

Obtenga información detallada sobre un clúster existente.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/clusters/{clusterName}
{
  "region": "string"
```



```
}
```

Cuerpo de la solicitud

clusterName

El nombre del clúster.

Tipo: String

Obligatorio: sí

region

La Región de AWS en la que se encuentra el clúster.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

Note

`failureReason` ha cambiado a `failures` a partir de AWS ParallelCluster versión 3.5.0.

```
{
  "clusterName": "string",
  "region": "string",
  "version": "string",
  "cloudFormationStackStatus": "CREATE_IN_PROGRESS",
  "clusterStatus": "CREATE_IN_PROGRESS",
  "scheduler": {
    "type": "string",
    "metadata": {
      "name": "string",
      "version": "string"
    }
  },
  "cloudformationStackArn": "string",
  "creationTime": "2019-08-24T14:15:22Z",
  "lastUpdatedTime": "2019-08-24T14:15:22Z",
```

```
"clusterConfiguration": {
  "url": "string"
},
"computeFleetStatus": "START_REQUESTED",
"tags": [
  {
    "key": "string",
    "value": "string"
  }
],
"headNode": {
  "instanceId": "string",
  "instanceType": "string",
  "launchTime": "2019-08-24T14:15:22Z",
  "privateIpAddress": "string",
  "publicIpAddress": "string",
  "state": "pending"
},
"failures": [
  {
    "failureCode": "string",
    "failureReason": "string"
  }
]
"loginNodes": {
  "status": "string",
  "address": "string",
  "scheme": "string",
  "healthyNodes": integer,
  "unhealthyNodes": integer
}
}
```

Cuerpo de respuesta

clusterName

El nombre del clúster.

Tipo: String

cloudformationStackArn

El nombre de recurso de Amazon (ARN) de la pila principal de CloudFormation.

Tipo: String

cloudformationStackStatus

El estado de la pila de CloudFormation.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| ROLLBACK_IN_PROGRESS | ROLLBACK_FAILED | ROLLBACK_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE_CLEANUP_IN_PROGRESS |
UPDATE_COMPLETE | UPDATE_ROLLBACK_IN_PROGRESS | UPDATE_ROLLBACK_FAILED |
UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS | UPDATE_ROLLBACK_COMPLETE

clusterConfiguration

url

La URL del archivo de configuración del clúster.

Tipo: String

clusterStatus

El estado del clúster.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE | UPDATE_FAILED

computeFleetStatus

El estado de la flota de computación.

Tipo: String

Valores válidos: START_REQUESTED | STARTING | RUNNING | PROTECTED |
STOP_REQUESTED | STOPPING | STOPPED | UNKNOWN | ENABLED | DISABLED

creationTime

Marca de tiempo de la creación del clúster.

Tipo: datetime

lastUpdatedTime

Marca de tiempo de la última actualización del clúster.

Tipo: datetime

region

La Región de AWS en la que se crea el clúster.

Tipo: String

etiquetas

La lista de etiquetas asociadas al clúster.

key

Nombre de la etiqueta.

Tipo: String

tag

Valor de la etiqueta.

Tipo: String

version

La versión de AWS ParallelCluster que se utiliza para crear el clúster.

Tipo: String

failures

La lista de errores cuando la pila de clústeres está en estado CREATE_FAILED.

failureCode

El código de error cuando la pila de clústeres está en estado CREATE_FAILED.

Tipo: String

failureReason

El motivo del error cuando la pila de clústeres está en estado CREATE_FAILED.

Tipo: String

head_node

El nodo principal del clúster.

instancetype

El identificador de la instancia de EC2.

Tipo: String

launchTime

El tipo de instancia de EC2.

Tipo: String

privateIpAddress

La hora en que se lanzó la instancia de EC2.

Tipo: datetime

publicIpAddress

La dirección IP privada del clúster.

Tipo: String

estado

La dirección IP pública del clúster.

Tipo: String

programador

El estado de la instancia de nodo principal.

Tipo: String

Valores válidos: `pending` | `running` | `shutting-down` | `terminated` | `stopping`
| `stopped`

metadatos

Los metadatos del programador.

nombre

El nombre del programador.

Tipo: String

version

La versión del programador.

Tipo: String

loginNodes**status**

El estado del nodo de inicio de sesión.

Tipo: String

Valores válidos: PENDING | FAILED | ACTIVE

address

La dirección del nodo de inicio de sesión.

Tipo: String

scheme

El esquema del nodo de inicio de sesión.

Tipo: String

scheme

El número de nodos en buen estado.

Tipo: número entero

scheme

El número de nodos en mal estado.

Tipo: número entero

type

El tipo de programador.

Tipo: String

Ejemplo

Python

Solicitud

```
$ describe_cluster(cluster_name_3x)
```

Respuesta 200

```
{
  'cloud_formation_stack_status': 'CREATE_COMPLETE',
  'cloudformation_stack_arn': 'arn:aws:cloudformation:us-east-1:123456789012:stack/
cluster_name_3x/16b49540-ae5-11ec-8e18-0ac1d712b241',
  'cluster_configuration': {
    'url': 'https://parallelcluster-....'
  },
  'cluster_name': 'cluster_name_3x',
  'cluster_status': 'CREATE_COMPLETE',
  'compute_fleet_status': 'RUNNING',
  'creation_time': datetime.datetime(2022, 3, 28, 22, 19, 9, 661000,
tzinfo=tzlocal()),
  'head_node': {
    'instance_id': 'i-abcdef01234567890',
    'instance_type': 't2.micro',
    'launch_time': datetime.datetime(2022, 3, 28, 22, 21, 56, tzinfo=tzlocal()),
    'private_ip_address': '172.31.56.3',
    'public_ip_address': '107.23.100.164',
    'state': 'running'
  },
  'last_updated_time': datetime.datetime(2022, 3, 28, 22, 19, 9, 661000,
tzinfo=tzlocal()),
  'region': 'us-east-1',
  'tags': [
    {
      'key': 'parallelcluster:version', 'value': '3.2.1'
    }
  ]
}
```

```
    }  
  ],  
  'version': '3.2.1'  
}
```

describeClusterInstances

Describe las instancias que pertenecen a un clúster.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/clusters/{clusterName}/instances  
{  
  "nextToken": "string",  
  "nodeType": "string",  
  "queueName": "string",  
  "region": "string"  
}
```

Cuerpo de la solicitud

clusterName

El nombre del clúster.

Tipo: String

Obligatorio: sí

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Obligatorio: no

nodeType

Filtra las instancias por tipo de nodo.

Tipo: String

Valores válidos: HeadNode, ComputeNode, LoginNode

Obligatorio: no

queueName

Filtra las instancias por nombre de cola.

Tipo: String

Obligatorio: no

region

La Región de AWS en la que se encuentra el clúster.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{
  "nextToken": "string",
  "instances": [
    {
      "instanceId": "string",
      "instanceType": "string",
      "launchTime": "2019-08-24T14:15:22Z",
      "privateIpAddress": "string",
      "publicIpAddress": "string",
      "state": "pending",
      "nodeType": "HeadNode",
      "queueName": "string"
    }
  ]
}
```

```
}  
]  
}
```

Cuerpo de respuesta

instances

La lista de instancias del clúster.

instanceld

El identificador de la instancia de EC2.

Tipo: String

instanceType

El tipo de instancia de EC2.

Tipo: String

launchTime

La hora en que se lanzó la instancia de EC2.

Tipo: datetime

nodeType

El tipo de nodo.

Tipo: String

Valores válidos: HeadNode, ComputeNode, LoginNode

publicIpAddress

La dirección IP pública del clúster.

Tipo: String

queueName

El nombre de la cola en la que la instancia de EC2 está respaldando un nodo.

Tipo: String

estado

El estado de la instancia de EC2 del nodo.

Tipo: String

Valores válidos: `pending` | `running` | `shutting-down` | `terminated` | `stopping` | `stopped`

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Ejemplo

Python

Solicitud

```
$ describe_cluster_instances(cluster_name_3x)
```

Respuesta 200

```
{
  'instances': [
    {
      'instance_id': 'i-abcdef01234567890',
      'instance_type': 't2.micro',
      'launch_time': datetime.datetime(2022, 3, 30, 14, 2, 7, tzinfo=tzlocal()),
      'node_type': 'HeadNode',
      'private_ip_address': '192.0.2.5',
      'public_ip_address': '198.51.100.180',
      'state': 'running'
    }
  ]
}
```

describeComputeFleet

Describe el estado de la flota de computación.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/clusters/{clusterName}/computeFleet
{
  "region": "string"
}
```

Cuerpo de la solicitud

clusterName

El nombre del clúster.

Tipo: String

Obligatorio: sí

region

La Región de AWS en la que se encuentra el clúster.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{
  "status": "START_REQUESTED",
  "lastStatusUpdatedTime": "2019-08-24T14:15:22Z"
}
```

Cuerpo de respuesta

status

Tipo: String

Valores válidos: START_REQUESTED | STARTING | RUNNING | PROTECTED | STOP_REQUESTED | STOPPING | STOPPED | UNKNOWN | ENABLED | DISABLED

lastStatusUpdatedTime

La marca de tiempo que representa la hora de la última actualización de estado.

Tipo: datetime

Ejemplo

Python

Solicitud

```
$ describe_compute_fleet(cluster_name_3x)
```

Respuesta 200

```
{
  'last_status_updated_time': datetime.datetime(2022, 3, 28, 22, 27, 14,
  tzinfo=tzlocal()),
  'status': 'RUNNING'
}
```

describelmage

Obtenga información detallada sobre una imagen existente.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)

- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/images/custom/{imageId}  
{  
  "region": "string"  
}
```

Cuerpo de la solicitud

imageId

El identificador de la imagen.

Tipo: String

Obligatorio: sí

region

La Región de AWS en la que se creó la imagen.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{  
  "imageId": "string",  
  "region": "string",  
  "version": "string",  
  "imageBuildStatus": "BUILD_IN_PROGRESS",  
  "imageBuildLogsArn": "string",  
  "cloudformationStackStatus": "CREATE_IN_PROGRESS",  
  "cloudformationStackStatusReason": "string",  
  "cloudformationStackArn": "string",  
  "creationTime": "2019-08-24T14:15:22Z",  
}
```

```
"cloudformationStackCreationTime": "2019-08-24T14:15:22Z",
"cloudformationStackTags": [
  {
    "key": "string",
    "value": "string"
  }
],
"imageConfiguration": {
  "url": "string"
},
"imagebuilderImageStatus": "PENDING",
"imagebuilderImageStatusReason": "string",
"ec2AmiInfo": {
  "amiId": "string",
  "tags": [
    {
      "key": "string",
      "value": "string"
    }
  ],
  "amiName": "string",
  "architecture": "string",
  "state": "PENDING",
  "description": "string"
}
}
```

Cuerpo de respuesta

imageId

El identificador de la imagen de la que se va a recuperar la información detallada.

Tipo: String

imageBuildStatus

El estado de creación de la imagen.

Tipo: String

Valores válidos: BUILD_IN_PROGRESS | BUILD_FAILED | BUILD_COMPLETE | DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE

imageConfiguration

url

La URL del archivo de configuración de la imagen.

Tipo: String

region

La Región de AWS en la que se crea la imagen.

Tipo: String

version

La versión de AWS ParallelCluster que se utiliza para crear la imagen.

Tipo: String

cloudformationStackArn

El nombre de recurso de Amazon (ARN) de la pila principal de CloudFormation.

Tipo: String

cloudformationStackCreationTime

La marca de tiempo en la que se creó la pila de CloudFormation.

Tipo: datetime

cloudformationStackStatus

El estado de la pila de CloudFormation.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| ROLLBACK_IN_PROGRESS | ROLLBACK_FAILED | ROLLBACK_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE_CLEANUP_IN_PROGRESS |
UPDATE_COMPLETE | UPDATE_ROLLBACK_IN_PROGRESS | UPDATE_ROLLBACK_FAILED |
UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS | UPDATE_ROLLBACK_COMPLETE

cloudformationStackStatusReason

El motivo del estado de la pila de CloudFormation.

Tipo: String

cloudformationStackTags

La lista de etiquetas de la pila de CloudFormation.

key

El nombre de la etiqueta.

Tipo: String

value

El valor de la etiqueta.

Tipo: String

creationTime

Marca de tiempo de la creación de la imagen.

Tipo: datetime

ec2AmiInfo

amild

El identificador de la AMI de EC2.

Tipo: String

amiName

El nombre de la AMI de EC2.

Tipo: String

architecture

La arquitectura de la AMI de EC2.

Tipo: String

estado

El estado de la AMI de EC2.

Tipo: String

Valores válidos: PENDING | AVAILABLE | INVALID | DEREGISTERED | TRANSIENT
| FAILED | ERROR

etiquetas

La lista de etiquetas de AMI de EC2.

key

Nombre de la etiqueta.

Tipo: String

value

Valor de la etiqueta.

Tipo: String

imagebuilderImageStatus

El estado de ImageBuilder.

Tipo: String

Valores válidos: PENDING | CREATING | BUILDING | TESTING | DISTRIBUTING |
INTEGRATING | AVAILABLE | CANCELLED | FAILED | DEPRECATED | DELETED

imagebuilderImageStatusReason

El motivo del estado de la imagen de ImageBuilder.

Tipo: String

imageBuildLogsArn

El nombre de recurso de Amazon (ARN) de los registros del proceso de creación de la imagen.

Tipo: String

Ejemplo

Python

Solicitud

```
$ describe_image(custom-image-id)
```

Respuesta 200

```
{
  'cloudformation_stack_arn': 'arn:aws:cloudformation:us-east-1:123456789012:stack/
custom-image-id/6accc570-b080-11ec-845e-0e2dc6386985',
  'cloudformation_stack_creation_time': datetime.datetime(2022, 3, 30, 23, 23, 33,
731000, tzinfo=tzlocal()),
  'cloudformation_stack_status': 'CREATE_IN_PROGRESS',
  'cloudformation_stack_tags': [
    {
      'key': 'parallelcluster:version', 'value': '3.2.1'
    },
    {
      'key': 'parallelcluster:image_name',
      'value': 'custom-image-id'
    },
    {
      'key': 'parallelcluster:custom-image-id',
      'value': 'custom-image-id'
    },
    {
      'key': 'parallelcluster:s3_bucket',
      'value': 'parallelcluster-abcdef01234567890-v1-do-not-delete'
    },
    {
      'key': 'parallelcluster:s3_image_dir',
      'value': 'parallelcluster/3.2.1/images/custom-image-id-1234567890abcdef0'
    },
    {
      'key': 'parallelcluster:build_log',
      'value': 'arn:aws:logs:us-east-1:123456789012:log-group:/aws/imagebuilder/
ParallelClusterImage-custom-image-id'
    },
    {
      'key': 'parallelcluster:build_config',
      'value': 's3://parallelcluster-abcdef01234567890-v1-do-not-delete/
parallelcluster/3.2.1/images/custom-image-id-1234567890abcdef0/configs/image-
config.yaml'
    }
  ],
  'image_build_logs_arn': 'arn:aws:logs:us-east-1:123456789012:log-group:/aws/
imagebuilder/ParallelClusterImage-alinux2-image',
  'image_build_status': 'BUILD_IN_PROGRESS',
  'image_configuration': {
```

```
'url': 'https://parallelcluster-abcdef01234567890-v1-do-not-
delete.s3.amazonaws.com/parallelcluster/3.2.1/images/custom-image-
id-1234567890abcdef0/configs/image-config.yaml?... '
},
'image_id': 'custom-image-id',
'imagebuilder_image_status': 'PENDING',
'region': 'us-east-1',
'version': '3.2.1'
}
```

getClusterLogEvents

Recupera los eventos asociados a un flujo de registro.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/clusters/{clusterName}/logstreams/{logStreamName}
{
  "endTime": datetime,
  "limit": float,
  "nextToken": "string",
  "region": "string",
  "startFromHead": boolean,
  "startTime": datetime
}
```

Cuerpo de la solicitud

clusterName

El nombre del clúster.

Tipo: String

Obligatorio: sí

logStreamName

El nombre del flujo de registro.

Tipo: String

Obligatorio: sí

endTime

El final del intervalo de tiempo, expresado en formato ISO 8601. No se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

Tipo: datetime

Formato: 2021-01-01T20:00:00Z

Obligatorio: no

limit

El número máximo de eventos de registro devueltos. Si no se especifica nada, este valor será el número máximo de eventos de registro que quepan en un tamaño de respuesta de 1 MB, hasta 10 000 eventos de registro.

Tipo: flotante

Obligatorio: no

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Obligatorio: no

region

La Región de AWS en la que se encuentra el clúster.

Tipo: String

Obligatorio: no

startFromHead

Si se establece en `true`, se devuelven primero los eventos de registro más antiguos. Si el valor es `false`, se devuelven primero los eventos de registro más recientes. El valor predeterminado es `false`.

Tipo: booleano

Obligatorio: no

startTime

El inicio del intervalo de tiempo, expresado en formato ISO 8601. Se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

Tipo: datetime

Formato: `2021-01-01T20:00:00Z`

Obligatorio: no

Sintaxis de la respuesta

```
{
  "nextToken": "string",
  "prevToken": "string",
  "events": [
    {
      "timestamp": "2019-08-24T14:15:22Z",
      "message": "string"
    }
  ]
}
```

Cuerpo de respuesta

events

La lista de eventos filtrados.

message

El mensaje del evento.

Tipo: String

timestamp

La marca temporal del evento.

Tipo: datetime

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

prevToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Ejemplo

Python

Solicitud

```
$ get_cluster_log_events(cluster_name_3x, log_stream_name=ip-192-0-2-26.i-  
abcdef01234567890.cfn-init)
```

Respuesta 200

```
"events": [  
  {  
    "message": "2022-09-22 16:40:15,127 [DEBUG] CloudFormation client initialized  
with endpoint https://cloudformation.us-east-1.amazonaws.com",  
    "timestamp": "2022-09-22T16:40:15.127Z"  
  },  
  {  
    "message": "2022-09-22 16:40:15,127 [DEBUG] Describing resource  
HeadNodeLaunchTemplate in stack cluster_name_3x",  
    "timestamp": "2022-09-22T16:40:15.127Z"  
  },  
  ...  
]
```

]

getClusterStackEvents

Recupera los eventos que están asociados a la pila de un clúster.

Note

A partir de la versión 3.6.0, AWS ParallelCluster utiliza pilas anidadas para crear los recursos asociados a las colas y los recursos de computación. La API de `GetClusterStackEvents` y el comando `pcluster get-cluster-stack-events` solo devuelven los eventos de la pila principal del clúster. Puede ver los eventos de la pila del clúster, incluidos los relacionados con las colas y los recursos de computación, en la consola CloudFormation.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/clusters/{clusterName}/stackevents
{
  "nextToken": "string",
  "region": "string"
}
```

Cuerpo de la solicitud

clusterName

El nombre del clúster.

Tipo: String

Obligatorio: sí

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Obligatorio: no

region

La Región de AWS en la que se encuentra el clúster.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{
  "nextToken": "string",
  "events": [
    {
      "stackId": "string",
      "eventId": "string",
      "stackName": "string",
      "logicalResourceId": "string",
      "physicalResourceId": "string",
      "resourceType": "string",
      "timestamp": "2019-08-24T14:15:22Z",
      "resourceStatus": "CREATE_IN_PROGRESS",
      "resourceStatusReason": "string",
      "resourceProperties": "string",
      "clientRequestToken": "string"
    }
  ]
}
```

Cuerpo de respuesta

events

La lista de eventos filtrados.

clientRequestToken

El token que se pasó a la acción y que generó este evento.

Tipo: String

eventId

El identificador exclusivo de este evento.

Tipo: String

logicalResourceId

El nombre lógico del recurso especificado en la plantilla.

Tipo: String

physicalResourceId

El nombre o identificador único que está asociado a la instancia física del recurso.

Tipo: String

resourceProperties

Un BLOB de las propiedades que se utilizan para crear el recurso.

Tipo: String

resourceStatus

El estado de los recursos.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE | DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE | DELETE_SKIPPED | UPDATE_IN_PROGRESS | UPDATE_FAILED | UPDATE_COMPLETE | IMPORT_FAILED | IMPORT_COMPLETE | IMPORT_IN_PROGRESS | IMPORT_ROLLBACK_IN_PROGRESS | IMPORT_ROLLBACK_FAILED | IMPORT_ROLLBACK_COMPLETE

resourceStatusReason

Un mensaje de éxito o error que está asociado al recurso.

Tipo: String

resourceType

El tipo de recurso.

Tipo: String

stackId

El nombre del identificador único de la instancia de la pila.

Tipo: String

stackName

El nombre que está asociado a una pila.

Tipo: String

timestamp

La hora a la que se actualizó el estado.

Tipo: datetime

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Ejemplo

Python

Solicitud

```
$ get_cluster_stack_events(cluster_name_3x)
```

Respuesta 200

```
{
```

```
'events': [
  {
    'event_id': '590b3820-b081-11ec-985e-0a7af5751497',
    'logical_resource_id': 'cluster_name_3x',
    'physical_resource_id': 'arn:aws:cloudformation:us-east-1:123456789012:stack/
cluster_name_3x/11a59710-b080-11ec-b8bd-129def1380e9',
    'resource_status': 'CREATE_COMPLETE',
    'resource_type': 'AWS::CloudFormation::Stack',
    'stack_id': 'arn:aws:cloudformation:us-east-1:123456789012:stack/
cluster_name_3x/11a59710-b080-11ec-b8bd-129def1380e9',
    'stack_name': 'cluster_name_3x',
    'timestamp': datetime.datetime(2022, 3, 30, 23, 30, 13, 268000,
tzinfo=tzlocal())
  },
  ...
]
```

getImageLogEvents

Recupera los eventos que están asociados a la creación de una imagen.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/images/custom/{imageId}/logstreams/{logStreamName}
{
  "endTime": datetime,
  "limit": float,
  "nextToken": "string",
  "region": "string",
  "startFromHead": boolean,
  "startTime": datetime
```

```
}
```

Cuerpo de la solicitud

imageId

El identificador de la imagen.

Tipo: String

Obligatorio: sí

logStreamName

El nombre del flujo de registro.

Tipo: String

Obligatorio: sí

endTime

El final del intervalo de tiempo, expresado en formato ISO 8601. No se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

Tipo: datetime

Formato: 2021-01-01T20:00:00Z

Obligatorio: no

limit

El número máximo de eventos de registro devueltos. Si no se especifica nada, este valor será el número máximo de eventos de registro que quepan en un tamaño de respuesta de 1 MB, hasta 10 000 eventos de registro.

Tipo: flotante

Obligatorio: no

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Obligatorio: no

region

La Región de AWS en la que se encuentra la imagen.

Tipo: String

Obligatorio: no

startFromHead

Si se establece en `true`, se devuelven primero los eventos de registro más antiguos. Si se establece en `false`, se devuelven primero los eventos de registro más recientes. El valor predeterminado es `false`.

Tipo: booleano

Obligatorio: no

startTime

El inicio del intervalo de tiempo, expresado en formato ISO 8601. Se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

Tipo: datetime

Formato: `2021-01-01T20:00:00Z`

Obligatorio: no

Sintaxis de la respuesta

```
{
  "nextToken": "string",
  "prevToken": "string",
  "events": [
    {
      "timestamp": "2019-08-24T14:15:22Z",
      "message": "string"
    }
  ]
}
```

```
}
```

Cuerpo de respuesta

events

Una lista de eventos filtrados.

message

El mensaje del evento.

Tipo: String

timestamp

La marca temporal del evento.

Tipo: datetime

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

prevToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Ejemplo

Python

Solicitud

```
$ get_image_log_events(image_id, log_stream_name=3.2.1/1)
```

Respuesta 200

```
"events": [  
  {
```

```
"message": "ExecuteBash: STARTED EXECUTION",
"timestamp": "2022-04-05T15:51:20.228Z"
},
{
"message": "ExecuteBash: Created temporary directory: /tmp/1234567890abcdef0",
"timestamp": "2022-04-05T15:51:20.228Z"
},
...
]
```

getImageStackEvents

Recupera los eventos que están asociados a la pila para la creación de una imagen.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/images/custom/{imageId}/stackevents
{
  "nextToken": "string",
  "region": "string"
}
```

Cuerpo de la solicitud

imageId

El identificador de la imagen.

Tipo: String

Obligatorio: sí

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Obligatorio: no

region

La Región de AWS en la que se encuentra la imagen.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{
  "nextToken": "string",
  "events": [
    {
      "stackId": "string",
      "eventId": "string",
      "stackName": "string",
      "logicalResourceId": "string",
      "physicalResourceId": "string",
      "resourceType": "string",
      "timestamp": "2019-08-24T14:15:22Z",
      "resourceStatus": "CREATE_IN_PROGRESS",
      "resourceStatusReason": "string",
      "resourceProperties": "string",
      "clientRequestToken": "string"
    }
  ]
}
```

Cuerpo de respuesta

events

Una lista de eventos filtrados.

clientRequestToken

El token que se pasó a la acción y que generó este evento.

Tipo: String

eventId

El identificador exclusivo de este evento.

Tipo: String

logicalResourceId

El nombre lógico del recurso especificado en la plantilla.

Tipo: String

physicalResourceId

El nombre o identificador único que está asociado a la instancia física del recurso.

Tipo: String

resourceProperties

Un BLOB de las propiedades que se utilizan para crear el recurso.

Tipo: String

resourceStatus

El estado de los recursos.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE | DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE | DELETE_SKIPPED | UPDATE_IN_PROGRESS | UPDATE_FAILED | UPDATE_COMPLETE | IMPORT_FAILED | IMPORT_COMPLETE | IMPORT_IN_PROGRESS | IMPORT_ROLLBACK_IN_PROGRESS | IMPORT_ROLLBACK_FAILED | IMPORT_ROLLBACK_COMPLETE

resourceStatusReason

Un mensaje de éxito o error que está asociado al recurso.

Tipo: String

resourceType

El tipo de recurso.

Tipo: String

stackId

El nombre del identificador único de la instancia de la pila.

Tipo: String

stackName

El nombre que está asociado a una pila.

Tipo: String

timestamp

La hora a la que se actualizó el estado.

Tipo: datetime

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Ejemplo

Python

Solicitud

```
$ get_image_stack_events(image_id)
```

Respuesta 200

```
{
  'events': [
    {
      'event_id': 'ParallelClusterImage-
CREATE_IN_PROGRESS-2022-03-30T23:26:33.499Z',
      'logical_resource_id': 'ParallelClusterImage',
```

```

    'physical_resource_id': 'arn:aws:imagebuilder:us-east-1:123456789012:image/
parallelclusterimage-alinux2-image/3.2.1/1',
    'resource_properties': {
        "InfrastructureConfigurationArn": "arn:aws:imagebuilder:us-
east-1:123456789012:infrastructure-configuration/parallelclusterimage-6accc570-
b080-11ec-845e-0e2dc6386985",
        "ImageRecipeArn": "arn:aws:imagebuilder:us-east-1:123456789012:image-recipe/
parallelclusterimage-alinux2-image/3.2.1",
        "DistributionConfigurationArn": "arn:aws:imagebuilder:us-
east-1:123456789012:distribution-configuration/parallelclusterimage-6accc570-
b080-11ec-845e-0e2dc6386985",
        "EnhancedImageMetadataEnabled": "false",
        "Tags": {
            "parallelcluster:image_name": "alinux2-
image", "parallelcluster:image_id": "alinux2-image"
        }
    },
    'resource_status': 'CREATE_IN_PROGRESS',
    'resource_status_reason': 'Resource creation Initiated',
    'resource_type': 'AWS::ImageBuilder::Image',
    'stack_id': 'arn:aws:cloudformation:us-east-1:123456789012:stack/alinux2-
image/6accc570-b080-11ec-845e-0e2dc6386985',
    'stack_name': 'alinux2-image',
    'timestamp': datetime.datetime(2022, 3, 30, 23, 26, 33, 499000,
tzinfo=tzlocal())
},
...
]
}

```

listClusters

Obtenga una lista de los clústeres existentes.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/clusters
{
  "clusterStatus": "string",
  "nextToken": "string",
  "region": "string"
}
```

Cuerpo de la solicitud

clusterStatus

Filtra por estado de clúster. El valor predeterminado es todos los clústeres.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | UPDATE_IN_PROGRESS |
UPDATE_COMPLETE | UPDATE_FAILED

Requerido: no

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Obligatorio: no

region

La Región de AWS de los clústeres.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{
```

```
"nextToken": "string",
"clusters": [
  {
    "clusterName": "string",
    "region": "string",
    "version": "string",
    "cloudformationStackArn": "string",
    "cloudformationStackStatus": "CREATE_IN_PROGRESS",
    "clusterStatus": "CREATE_IN_PROGRESS",
    "scheduler": {
      "type": "string",
      "metadata": {
        "name": "string",
        "version": "string"
      }
    }
  }
]
}
```

Cuerpo de respuesta

clústeres

cloudformationStackArn

El nombre de recurso de Amazon (ARN) de la pila principal de CloudFormation.

Tipo: String

cloudformationStackStatus

El estado de la pila de CloudFormation.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| ROLLBACK_IN_PROGRESS | ROLLBACK_FAILED | ROLLBACK_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_COMPLETE | UPDATE_ROLLBACK_IN_PROGRESS |
UPDATE_ROLLBACK_FAILED | UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_ROLLBACK_COMPLETE

clusterName

El nombre del clúster.

Tipo: String

clusterStatus

El estado del clúster.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE | UPDATE_FAILED

programador**metadatos**

Los metadatos del programador.

nombre

El nombre del programador.

Tipo: String

version

La versión del programador.

Tipo: String

type

El tipo de programador.

Tipo: String

region

La Región de AWS en la que se crea el clúster.

Tipo: String

version

La versión de AWS ParallelCluster que se utiliza para crear el clúster.

Tipo: String

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Ejemplo

Python

Solicitud

```
$ list_clusters()
```

Respuesta 200

```
{
  'clusters':
  [
    {
      'cloudformation_stack_arn': 'arn:aws:cloudformation:us-
east-1:123456789012:stack/cluster_name_3x/16b49540-ae5-11ec-8e18-0ac1d712b241',
      'cloudformation_stack_status': 'CREATE_COMPLETE',
      'cluster_name': 'cluster_name_3x',
      'cluster_status': 'CREATE_COMPLETE',
      'region': 'us-east-1',
      'version': '3.2.1'
    },
    ...
  ]
}
```

listClusterLogStreams

Obtenga la lista de flujos de registro que están asociados a un clúster.

Temas

- [Sintaxis de la solicitud](#)

- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/clusters/{clusterName}/logstreams
{
  "filters": [ "string" ],
  "nextToken": "string",
  "region": "string"
}
```

Cuerpo de la solicitud

clusterName

El nombre del clúster.

Tipo: String

Obligatorio: sí

filters

Filtra los flujos de registro.

Los filtros aceptados son:

- private-dns-name: forma abreviada del nombre de DNS privado de la instancia (por ejemplo, ip-10-0-0-101).
- node-type: valor válido: HeadNode.

Tipo: matriz de cadenas únicas

Formato: Name=a, Values=1 Name=b, Values=2, 3

Obligatorio: no

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Obligatorio: no

region

La Región de AWS en la que se encuentra el clúster.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{
  "nextToken": "string",
  "logStreams": [
    {
      "logStreamName": "string",
      "creationTime": "2019-08-24T14:15:22Z",
      "firstEventTimestamp": "2019-08-24T14:15:22Z",
      "lastEventTimestamp": "2019-08-24T14:15:22Z",
      "lastIngestionTime": "2019-08-24T14:15:22Z",
      "uploadSequenceToken": "string",
      "logStreamArn": "string"
    }
  ]
}
```

Cuerpo de respuesta

logStreams

Una lista de flujos de registro.

creationTime

La hora en la que se creó el flujo.

Tipo: datetime

firstEventTimestamp

La hora del primer evento del flujo.

Tipo: datetime

lastEventTimestamp

La hora del último evento del flujo. El valor de lastEventTime se actualiza en función de la coherencia eventual. Por lo general, se actualiza en menos de una hora desde la incorporación, pero en raras ocasiones puede llevar más tiempo.

Tipo: datetime

lastIngestionTime

La hora de la última incorporación.

Tipo: datetime

logStreamArn

El nombre de recurso de Amazon (ARN) del flujo de registro.

Tipo: String

logStreamName

El nombre del flujo de registro.

Tipo: String

uploadSequenceToken

El token de la secuencia.

Tipo: String

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Ejemplo

Python

Solicitud

```
$ list_cluster_log_streams(cluster_name_3x)
```

Respuesta 200

```
{
  'log_streams': [
    {
      'creation_time': datetime.datetime(2022, 3, 30, 14, 7, 34, 354000,
tzinfo=tzlocal()),
      'first_event_timestamp': datetime.datetime(2022, 3, 30, 14, 6, 41, 444000,
tzinfo=tzlocal()),
      'last_event_timestamp': datetime.datetime(2022, 3, 30, 14, 25, 55, 462000,
tzinfo=tzlocal()),
      'last_ingestion_time': datetime.datetime(2022, 3, 30, 14, 49, 50, 62000,
tzinfo=tzlocal()),
      'log_stream_arn': 'arn:aws:logs:us-east-1:123456789012:log-group:/aws/
parallelcluster/cluster_name_3x:log-stream:ip-192-0-2-26.i-abcdef01234567890.cfn-
init',
      'log_stream_name': 'ip-192-0-2-26.i-abcdef01234567890.cfn-init',
      ...
      'upload_sequence_token': '####'
    },
    ...
  ]
}
```

listImageLogStreams

Obtenga la lista de flujos de registro que está asociada a una imagen.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/images/custom/{imageId}/logstreams
{
```

```
"nextToken": "string",  
"region": "string"  
}
```

Cuerpo de la solicitud

imageId

El identificador de la imagen.

Tipo: String

Obligatorio: sí

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Obligatorio: no

region

La Región de AWS en la que se encuentra la imagen.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{  
  "nextToken": "string",  
  "logStreams": [  
    {  
      "logStreamName": "string",  
      "creationTime": "2019-08-24T14:15:22Z",  
      "firstEventTimestamp": "2019-08-24T14:15:22Z",  
      "lastEventTimestamp": "2019-08-24T14:15:22Z",  
      "lastIngestionTime": "2019-08-24T14:15:22Z",  
      "uploadSequenceToken": "string",  
      "logStreamArn": "string"  
    }  
  ]  
}
```

```
}  
]  
}
```

Cuerpo de respuesta

logStreams

Una lista de flujos de registro.

creationTime

La hora en la que se creó el flujo.

Tipo: datetime

firstEventTimestamp

La hora del primer evento del flujo.

Tipo: datetime

lastEventTimestamp

La hora del último evento del flujo. El valor de lastEventTime se actualiza en función de la coherencia eventual. Por lo general, se actualiza en menos de una hora desde la incorporación, pero en raras ocasiones puede llevar más tiempo.

Tipo: datetime

lastIngestionTime

La hora de la última incorporación.

Tipo: datetime

logStreamArn

El nombre de recurso de Amazon (ARN) del flujo de registro.

Tipo: String

logStreamName

El nombre del flujo de registro.

Tipo: String

uploadSequenceToken

El token de la secuencia.

Tipo: String

next_token

Un token que se usa para las solicitudes paginadas.

Tipo: String

Ejemplo

Python

Solicitud

```
$ list_image_log_streams(custom-image-id)
```

Respuesta 200

```
{
  'log_streams': [
    {
      'creation_time': datetime.datetime(2022, 3, 29, 20, 29, 24, 875000,
tzinfo=tzlocal()),
      'first_event_timestamp': datetime.datetime(2022, 3, 29, 20, 29, 24, 775000,
tzinfo=tzlocal()),
      'last_event_timestamp': datetime.datetime(2022, 3, 29, 20, 38, 23, 944000,
tzinfo=tzlocal()),
      'last_ingestion_time': datetime.datetime(2022, 3, 29, 20, 51, 56, 26000,
tzinfo=tzlocal()),
      'log_stream_arn': 'arn:aws:logs:us-east-1:123456789012:log-group:/aws/
imagebuilder/ParallelClusterImage-alinux2-image:log-stream:3.2.1/1',
      'log_stream_name': '3.2.1/1',
      'upload_sequence_token': '####'
    },
    ...
  ]
}
```

listImages

Obtenga la lista de imágenes personalizadas existentes.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /images/custom
{
  "imageStatus": "string",
  "nextToken": "string",
  "region": "string"
}
```

Cuerpo de la solicitud

imageStatus

Filtre las imágenes según el estado proporcionado.

Tipo: String

Valores válidos: AVAILABLE | PENDING | FAILED

Obligatorio: sí

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Obligatorio: no

region

La Región de AWS en la que se encuentran las imágenes.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{
  "nextToken": "string",
  "images": [
    {
      "imageId": "string",
      "ec2AmiInfo": {
        "amiId": "string"
      },
      "region": "string",
      "version": "string",
      "cloudformationStackArn": "string",
      "imageBuildStatus": "BUILD_IN_PROGRESS",
      "cloudformationStackStatus": "CREATE_IN_PROGRESS"
    }
  ]
}
```

Cuerpo de respuesta

images

Una lista de imágenes.

cloudformationStackArn

El nombre de recurso de Amazon (ARN) de la pila principal de CloudFormation.

Tipo: String

cloudformationStackStatus

El estado de la pila de CloudFormation.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| ROLLBACK_IN_PROGRESS | ROLLBACK_FAILED | ROLLBACK_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_COMPLETE | UPDATE_ROLLBACK_IN_PROGRESS |
UPDATE_ROLLBACK_FAILED | UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_ROLLBACK_COMPLETE

ec2AmiInfo

ami-id

El identificador de la AMI de EC2.

Tipo: String

imageBuildStatus

El estado de creación de la imagen.

Valores válidos: BUILD_IN_PROGRESS | BUILD_FAILED | BUILD_COMPLETE |
DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE

Tipo: String

imageId

El identificador de la imagen.

Tipo: String

region

La Región de AWS en la que se crea la imagen.

Tipo: String

version

La versión de AWS ParallelCluster que se utiliza para crear la imagen.

Tipo: String

nextToken

Un token que se usa para las solicitudes paginadas.

Tipo: String

Ejemplo

Python

Solicitud

```
$ list_images("AVAILABLE")
```

Respuesta 200

```
{
  'images': [
    {
      'ec2_ami_info': {
        'ami_id': 'ami-abcdef01234567890'
      },
      'image_build_status': 'BUILD_COMPLETE',
      'image_id': 'custom-image',
      'region': 'us-east-1',
      'version': '3.2.1'
    }
  ]
}
```

listOfficialImages

Obtenga la lista de imágenes oficiales de AWS ParallelCluster.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
GET /v3/images/official
```

```
{
  "architecture": "string",
  "os": "string",
  "region": "string"
}
```

Cuerpo de la solicitud

architecture

Filtra por arquitectura. El valor predeterminado es no filtrar.

Tipo: String

Valores válidos: x86_64 | arm64

Obligatorio: no

os

Filtra por distribución del sistema operativo. El valor predeterminado es no filtrar.

Tipo: String

Valores válidos: alinux2 | centos7 | ubuntu2204 | ubuntu2004 | rhe18

Obligatorio: no

region

La Región de AWS en la que aparecen las imágenes oficiales.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{
  "images": [
    {
      "architecture": "string",
      "amiId": "string",
      "name": "string",
    }
  ]
}
```

```
    "os": "string",
    "version": "string"
  }
]
```

Cuerpo de respuesta

images

amild

Es el ID de la AMI.

Tipo: String

architecture

La arquitectura de AMI.

Tipo: String

nombre

El nombre de la AMI.

Tipo: String

os

El sistema operativo de la AMI.

Tipo: String

version

La versión de AWS ParallelCluster.

Tipo: String

Ejemplo

Python

Solicitud

```
$ list_official_images()
```

Respuesta 200

```
{
  'images': [
    {
      'ami_id': 'ami-015cfefb4e0d6306b2',
      'architecture': 'x86_64',
      'name': 'aws-parallelcluster-3.2.1-ubuntu-2004-lts-hvm-x86_64-202202261505 '
      '2022-02-26T15-08-34.759Z',
      'os': 'ubuntu2004',
      'version': '3.2.1'
    },
    ...
  ]
}
```

updateCluster

Actualiza el clúster.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
PUT /v3/clusters/{clusterName}
{
  "clusterConfiguration": "string",
  "dryrun": boolean,
  "forceUpdate": boolean,
  "region": "string",
  "suppressValidators": "string",
```

```
"validationFailureLevel": "string"  
}
```

Cuerpo de la solicitud

clusterConfiguration

La configuración del clúster como documento YAML.

Obligatorio: sí

clusterName

El nombre del clúster.

Tipo: String

Obligatorio: sí

dryrun

Si se establece en `true`, solo realiza la validación de la solicitud sin crear ningún recurso. Utilice este parámetro para validar la configuración del clúster y los requisitos de actualización. El valor predeterminado es `false`.

Tipo: booleano

Obligatorio: no

forceUpdate

Si se establece en `true`, ignora los errores de validación de la actualización y fuerza la actualización. El valor predeterminado es `false`.

Tipo: booleano

Obligatorio: no

region

La Región de AWS en la que se encuentra el clúster.

Tipo: String

Obligatorio: no

suppressValidators

Identifica uno o más validadores de configuración para suprimirlos.

Tipo: String

Formato: (ALL | type: [A-Za-z0-9]+)

Obligatorio: no

Ejemplos de valores válidos: currentValue, requestedValue, message

validationFailureLevel

El nivel mínimo de validación para provocar un error en la actualización.

Tipo: String

Valores válidos: INFO | WARNING | ERROR

Obligatorio: no

Sintaxis de la respuesta

```
{
  "cluster": {
    "clusterName": "string",
    "region": "string",
    "version": "string",
    "cloudformationStackArn": "string",
    "cloudformationStackStatus": "UPDATE_IN_PROGRESS",
    "clusterStatus": "UPDATE_IN_PROGRESS",
    "scheduler": {
      "type": "string",
      "metadata": {
        "name": "string",
        "version": "string"
      }
    }
  },
  "validationMessages": [
    {
      "id": "string",
```



```
    "type": "string",
    "level": "INFO",
    "message": "string"
  }
],
"changeSet": [
  {
    "parameter": "string",
    "currentValue": "string",
    "requestedValue": "string"
  }
]
}
```

Cuerpo de respuesta

changeSet

El conjunto de cambios para la actualización del clúster.

currentValue

El valor actual del parámetro que se va a actualizar.

Tipo: String

parameter

El parámetro que se va a actualizar.

Tipo: String

requestedValue

El valor solicitado para el parámetro que se va a actualizar.

Tipo: String

clúster

cloudformationStackArn

El nombre de recurso de Amazon (ARN) de la pila principal de CloudFormation.

Tipo: String

cloudformationStackStatus

El estado de la pila de CloudFormation.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| ROLLBACK_IN_PROGRESS | ROLLBACK_FAILED | ROLLBACK_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_COMPLETE | UPDATE_ROLLBACK_IN_PROGRESS |
UPDATE_ROLLBACK_FAILED | UPDATE_ROLLBACK_COMPLETE_CLEANUP_IN_PROGRESS
| UPDATE_ROLLBACK_COMPLETE

clusterName

El nombre del clúster.

Tipo: String

clusterStatus

El estado del clúster.

Tipo: String

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE
| DELETE_IN_PROGRESS | DELETE_FAILED | DELETE_COMPLETE |
UPDATE_IN_PROGRESS | UPDATE_COMPLETE | UPDATE_FAILED

region

La Región de AWS en la que se crea el clúster.

Tipo: String

programador

metadatos

Los metadatos del programador.

nombre

El nombre del programador.

Tipo: String

version

La versión del programador.

Tipo: String

type

El tipo de programador.

Tipo: String

version

La versión de AWS ParallelCluster que se utiliza para crear el clúster.

Tipo: String

validationMessages

Una lista de mensajes con un nivel de validación inferior a `validationFailureLevel`. La lista de mensajes se recopila durante la validación de la configuración.

id

El identificador del validador.

Tipo: String

level

El nivel de validación.

Tipo: String

Valores válidos: INFO | WARNING | ERROR

message

El mensaje de validación.

Tipo: String

type

El tipo de validador.

Tipo: String

Ejemplo

Python

Solicitud

```
$ update_cluster(cluster_name_3x, path/config-file.yaml)
```

Respuesta 200

```
{
  'change_set': [
    {
      'current_value': '10',
      'parameter':
'Scheduling.SlurmQueues[queue1].ComputeResources[t2micro].MaxCount',
      'requested_value': '15'
    }
  ],
  'cluster': {
    'cloudformation_stack_arn': 'arn:aws:cloudformation:us-
east-1:123456789012:stack/test-api-cluster/e0462730-50b5-11ed-99a3-0a5ddc4a34c7',
    'cloudformation_stack_status': 'UPDATE_IN_PROGRESS',
    'cluster_name': 'cluster-3x',
    'cluster_status': 'UPDATE_IN_PROGRESS',
    'region': 'us-east-1',
    'scheduler': {
      'type': 'slurm'
    },
    'version': '3.2.1'
  }
}
```

updateComputeFleet

Actualice el estado de la flota de computación del clúster.

Temas

- [Sintaxis de la solicitud](#)
- [Cuerpo de la solicitud](#)
- [Sintaxis de la respuesta](#)
- [Cuerpo de respuesta](#)
- [Ejemplo](#)

Sintaxis de la solicitud

```
PATCH /v3/clusters/{clusterName}/computefleet
{
  "status": "string",
  "region": "string"
}
```

Cuerpo de la solicitud

clusterName

El nombre del clúster.

Tipo: String

Obligatorio: sí

status

El estado de la flota de computación.

Tipo: String

Valores válidos: START_REQUESTED | STOP_REQUESTED | ENABLED | DISABLED

Obligatorio: sí

region

La Región de AWS en la que se encuentra el clúster.

Tipo: String

Obligatorio: no

Sintaxis de la respuesta

```
{
  "status": "START_REQUESTED",
  "lastStatusUpdateTime": "2019-08-24T14:15:22Z"
}
```

Cuerpo de respuesta

status

El estado de la flota de computación.

Tipo: String

Valores válidos: START_REQUESTED | STARTING | RUNNING | PROTECTED | STOP_REQUESTED | STOPPING | STOPPED | UNKNOWN | ENABLED | DISABLED

lastStatusUpdateTime

La marca de tiempo que representa la hora de la última actualización de estado.

Tipo: datetime

Ejemplo

Python

Solicitud

```
$ update_compute_fleet(cluster_name_3x, "START_REQUESTED")
```

Respuesta 200

```
{
  'last_status_updated_time': datetime.datetime(2022, 3, 28, 22, 27, 14,
  tzinfo=tzlocal()),
```

```
'status': 'START_REQUESTED'  
}
```

API de la biblioteca Python de AWS ParallelCluster

A partir de AWS ParallelCluster versión 3.5.0, puede acceder a AWS ParallelCluster con la biblioteca Python de AWS ParallelCluster. Puede acceder a la biblioteca de AWS ParallelCluster en su entorno de `pcluster` o desde un tiempo de ejecución de AWS Lambda. Aprenda a acceder a la API de AWS ParallelCluster mediante la biblioteca Python de AWS ParallelCluster. La biblioteca Python de AWS ParallelCluster ofrece la misma funcionalidad que ofrece la API de AWS ParallelCluster.

Las operaciones y los parámetros de la biblioteca Python de AWS ParallelCluster reflejan los de los parámetros de la API cuando se convierten a `snake_case` sin mayúsculas.

Temas

- [Autorización de las bibliotecas Python de AWS ParallelCluster](#)
- [Instalación de la biblioteca Python de AWS ParallelCluster](#)
- [Operaciones de la API del clúster](#)
- [Computación de las operaciones de la API de flota](#)
- [Operaciones de registro de clústeres y pilas](#)
- [Operaciones de la API de imagen](#)
- [Operaciones de registro de imágenes y pilas](#)
- [Ejemplo](#)
- [AWS Lambda para la biblioteca Python de AWS ParallelCluster](#)

Autorización de las bibliotecas Python de AWS ParallelCluster

Especifique las credenciales mediante cualquiera de las formas estándar válidas para boto3. Para obtener más información, consulte la [documentación de boto3](#).

Instalación de la biblioteca Python de AWS ParallelCluster

1. Instale la CLI de `pcluster` versión 3.5.0 o posterior siguiendo las instrucciones que se proporcionan en [Con AWS ParallelCluster figuración](#).

2. Importe el módulo de `pcluster` y comience a usar la biblioteca, tal y como se muestra en el ejemplo siguiente:

```
import pcluster.lib as pc
pc.create_cluster(cluster_name="mycluster", cluster_configuration="config.yaml")
```

Operaciones de la API del clúster

Temas

- [list_clusters](#)
- [create_cluster](#)
- [delete_cluster](#)
- [describe_cluster](#)
- [update_cluster](#)

list_clusters

```
list_clusters(region, next_token, cluster_status)
```

Obtenga la lista de los clústeres existentes.

Parámetros:

region

Muestra los clústeres implementados en una Región de AWS determinada.

next_token

El token que se utilizará para las solicitudes paginadas.

cluster_status

Filtra por estado del clúster. El valor predeterminado es mostrar todos los clústeres.

Valores válidos: CREATE_IN_PROGRESS | CREATE_FAILED | CREATE_COMPLETE |
DELETE_IN_PROGRESS | DELETE_FAILED | UPDATE_IN_PROGRESS | UPDATE_COMPLETE |
UPDATE_FAILED

create_cluster

```
create_cluster(cluster_name, cluster_configuration, region, suppress_validators,
               validation_failure_level, dry_run, rollback_on_failure, wait)
```

Cree un clúster en una región determinada.

Parámetros:

cluster_name (obligatorio)

El nombre del clúster.

cluster_configuration (obligatorio)

La configuración del clúster como un tipo de datos de Python.

region

La Región de AWS del clúster.

suppress_validators

Identifica uno o más validadores de configuración de clústeres para suprimirlos.

Formato: (ALL | type:[A-Za-z0-9]+)

validation_failure_level

El nivel mínimo de validación que provoca un error en la creación del clúster. El valor predeterminado es ERROR.

Valores válidos: INFO | WARNING | ERROR.

dry_run

Realiza la validación de la solicitud sin crear ningún recurso. Puede usarlo para validar la configuración del clúster. El valor predeterminado es False.

rollback_on_failure

Si se establece en True, AWS ParallelCluster inicia automáticamente una reversión de la pila del clúster en caso de que se produzcan errores. El valor predeterminado es True.

wait

Si se establece en True, AWS ParallelCluster espera a que se complete la operación. El valor predeterminado es False.

delete_cluster

```
delete_cluster(cluster_name, region, wait)
```

Elimine un clúster de una región determinada.

Parámetros:

cluster_name (obligatorio)

El nombre del clúster.

region

La Región de AWS del clúster.

wait

Si se establece en `True`, espera a que se complete la operación. El valor predeterminado es `False`.

describe_cluster

```
describe_cluster(cluster_name, region)
```

Obtenga información detallada sobre un clúster existente.

Parámetros:

cluster_name (obligatorio)

El nombre del clúster.

region

La Región de AWS del clúster.

update_cluster

```
update_cluster(cluster_name, cluster_configuration, suppress_validators,  
validation_failure_level, region, force_update, dry_run, wait)
```

Actualice un clúster de una región determinada.

Parámetros:

cluster_name (obligatorio)

El nombre del clúster.

cluster_configuration (obligatorio)

La configuración del clúster como un tipo de datos de Python.

suppress_validators

Identifica uno o más validadores de configuración de clústeres para suprimirlos.

Formato: (ALL | type:[A-Za-z0-9]+)

validation_failure_level

El nivel mínimo de validación que provoca un error en la actualización del clúster. El valor predeterminado es ERROR.

Valores válidos: INFO | WARNING | ERROR

region

La Región de AWS del clúster.

dry_run

Realiza la validación de la solicitud sin crear ni actualizar ningún recurso. Puede usarlo para validar la configuración del clúster. El valor predeterminado es False.

force_update

Si se establece en True, fuerza la actualización ignorando los errores de validación de la actualización. El valor predeterminado es False.

wait

Si se establece en True, espera a que se complete la operación. El valor predeterminado es False.

Computación de las operaciones de la API de flota

Temas

- [describe_compute_fleet](#)
- [update_compute_fleet](#)
- [delete_cluster_instances](#)
- [describe_cluster_instances](#)

describe_compute_fleet

```
describe_compute_fleet(cluster_name, region)
```

Describe el estado de la flota de computación de un clúster para un clúster determinado.

Parámetros:

cluster_name (obligatorio)

El nombre del clúster.

region

Describe el estado de la flota de computación de un clúster implementado en una Región de AWS determinada.

update_compute_fleet

```
update_compute_fleet(cluster_name, status, region)
```

Actualice el estado de la flota de computación del clúster.

Parámetros:

cluster_name (obligatorio)

El nombre del clúster.

status (obligatorio)

El estado al que se va a actualizar.

Valores válidos: START_REQUESTED | STOP_REQUESTED | ENABLED | DISABLED

region

La Región de AWS del clúster.

delete_cluster_instances

```
delete_cluster_instances(cluster_name, region, force)
```

Elimine un clúster de una región determinada.

Parámetros:

cluster_name (obligatorio)

El nombre del clúster.

region

La Región de AWS del clúster.

force

Si se establece en `True`, fuerza la eliminación cuando no se encuentra el clúster con el `cluster_name` indicado. El valor predeterminado es `False`.

describe_cluster_instances

```
describe_cluster_instances(cluster_name, region, next_token, node_type, queue_name)
```

Describa las instancias de un clúster.

Parámetros:

cluster_name (obligatorio)

El nombre del clúster.

region

La Región de AWS del clúster.

next_token

El token que se utilizará para las solicitudes paginadas.

node_type

Filtra las instancias por node_type.

Valores válidos: HeadNode | ComputeNode

queue_name

Filtra las instancias por nombre de cola.

Operaciones de registro de clústeres y pilas

Temas

- [list_cluster_log_streams](#)
- [get_cluster_log_events](#)
- [get_cluster_stack_events](#)

list_cluster_log_streams

```
list_cluster_log_streams(cluster_name, region, filters, next_token)
```

Muestra los flujos de registros de un clúster determinado.

Parámetros:

cluster_name (obligatorio)

El nombre del clúster.

region

La Región de AWS del clúster.

filters

Filtra los flujos de registro del clúster.

Formato: 'Name=a,Values=1 Name=b,Values=2,3'

Filtros aceptados:**code-dns-name**

La forma abreviada del nombre de DNS privado de la instancia; por ejemplo, `ip-10-0-0-101`.

Tipo de nodo

El tipo de nodo.

Valores válidos: `HeadNode`

next_token

El token que se utilizará para las solicitudes paginadas.

get_cluster_log_events

```
get_cluster_log_events(cluster_name, log_stream_name, region, next_token,
start_from_head, limit, start_time, end_time)
```

Obtenga los eventos de registro para un clúster y un flujo de registro determinados.

Parámetros:**cluster_name** (obligatorio)

El nombre del clúster.

log_stream_name (obligatorio)

El nombre del flujo de registro.

region

La Región de AWS del clúster.

next_token

El token que se utilizará para las solicitudes paginadas.

start_from_head

Si se establece en `True`, AWS ParallelCluster devuelve primero los eventos de registro más antiguos. Si se establece en `False`, devuelve primero los eventos de registro más recientes. El valor predeterminado es `False`.

limit

El número máximo de eventos de registro devueltos. Si no especifica un valor, el máximo es el número de registros que caben en un tamaño de respuesta de 1 MB, hasta 10 000 eventos de registro.

start_time

El inicio del intervalo de tiempo del registro de eventos, expresado en formato ISO 8601; por ejemplo, '2021-01-01T20:00:00Z'. Se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

end_time

El final del intervalo de tiempo del registro de eventos, expresado en formato ISO 8601; por ejemplo, '2021-01-01T20:00:00Z'. No se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

get_cluster_stack_events

```
get_cluster_stack_events(cluster_name, region, next_token)
```

Obtenga eventos de pila para un clúster determinado.

Parámetros:

cluster_name (obligatorio)

El nombre del clúster.

region

La Región de AWS del clúster.

next_token

El token que se utilizará para las solicitudes paginadas.

Operaciones de la API de imagen

Temas

- [list_images](#)
- [build_image](#)
- [delete_image](#)
- [describe_image](#)

list_images

```
list_images(image_status, region, next_token)
```

Obtenga la lista de imágenes existentes.

Parámetros:

image_status (obligatorio)

Filtra por estado de imagen.

Valores válidos: AVAILABLE | PENDING | FAILED

region

Muestra las imágenes creadas en una Región de AWS determinada.

next_token

El token que se utilizará para las solicitudes paginadas.

build_image

```
build_image(image_configuration, image_id, suppress_validators,  
validation_failure_level, dry_run, rollback_on_failure, region)
```

Crea una imagen personalizada de AWS ParallelCluster en una región determinada.

Parámetros:

image_configuration (obligatorio)

La configuración de la imagen como datos de Python.

image_id (obligatorio)

El ID de imagen.

suppress_validators

Identifica uno o más validadores de configuración de imágenes para suprimirlos.

Formato: (ALL | type:[A-Za-z0-9]+)

validation_failure_level

El nivel mínimo de validación que provoca un error en la creación de la imagen. El valor predeterminado es ERROR.

Valores válidos: INFO | WARNING | ERROR

dry_run

Si se establece en `True`, AWS ParallelCluster realiza la validación de la solicitud sin crear ningún recurso. Puede usarlo para validar la configuración de la imagen. El valor predeterminado es `False`.

rollback_on_failure

Si se establece en `True`, AWS ParallelCluster inicia automáticamente una reversión de la pila de la imagen en caso de que se produzcan errores. El valor predeterminado es `False`.

region

La imagen Región de AWS.

delete_image

```
delete_image(image_id, region, force)
```

Elimine una imagen de una región determinada.

Parámetros:

image_id (obligatorio)

El ID de imagen.

region

La imagen Región de AWS.

force

Si se establece en `True`, AWS ParallelCluster fuerza la eliminación si las instancias utilizan la AMI o si la AMI es compartida. El valor predeterminado es `False`.

describe_image

```
describe_image(image_id, region)
```

Obtenga información detallada sobre una imagen existente.

Parámetros:

image_id (obligatorio)

El ID de imagen.

region

La imagen Región de AWS.

Operaciones de registro de imágenes y pilas

Temas

- [list_image_log_streams](#)
- [get_image_log_events](#)
- [get_image_stack_events](#)
- [list_official_images](#)

list_image_log_streams

```
list_image_log_streams(image_id, region, next_token)
```

Muestre los flujos de registro de una imagen.

Parámetros:

image_id (obligatorio)

El ID de imagen.

region

La imagen Región de AWS.

next_token

El token que se utilizará para las solicitudes paginadas.

get_image_log_events

```
get_image_log_events(image_id, log_stream_name, region, next_token, start_from_head,
                    limit, start_time, end_time)
```

Obtenga los eventos de registro para una imagen y un flujo de registro determinados.

Parámetros:

image_id (obligatorio)

El ID de imagen.

log_stream_name (obligatorio)

El nombre del flujo de registro.

region

La imagen Región de AWS.

next_token

El token que se utilizará para las solicitudes paginadas.

start_from_head

Si se establece en `True`, AWS ParallelCluster devuelve primero los eventos de registro más antiguos. Si se establece en `False`, devuelve primero los eventos de registro más recientes. El valor predeterminado es `False`.

limit

El número máximo de eventos de registro devueltos. Si no especifica un valor, el máximo es el número de registros que caben en un tamaño de respuesta de 1 MB, hasta 10 000 eventos de registro.

start_time

El inicio del intervalo de tiempo del registro de eventos, expresado en formato ISO 8601; por ejemplo, '2021-01-01T20:00:00Z'. Se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

end_time

El final del intervalo de tiempo del registro de eventos, expresado en formato ISO 8601; por ejemplo, '2021-01-01T20:00:00Z'. No se incluyen los eventos con una marca de tiempo igual o posterior a esta hora.

get_image_stack_events

```
get_image_stack_events(image_id, region, next_token)
```

Obtenga los eventos de la pila para una imagen determinada.

Parámetros:

image_id (obligatorio)

El ID de imagen.

region

La imagen Región de AWS.

next_token

El token que se utilizará para las solicitudes paginadas.

list_official_images

```
list_official_images(region, os, architecture)
```

Obtenga la lista de imágenes oficiales de AWS ParallelCluster.

Parámetros:

region

La imagen Región de AWS.

os

Filtra por distribución de sistema operativo. El valor predeterminado es no filtrar.

architecture

Filtra por arquitectura. El valor predeterminado es no filtrar.

Ejemplo

Temas

- [Crear un clúster](#)

Crear un clúster

Al ejecutar el siguiente script de ejemplo, con las entradas indicadas almacenadas en su entorno, se crea un clúster. La configuración del clúster se crea como un tipo de datos de Python en función de la [documentación de configuración del clúster](#).

```
import os
import pprint
import pcluster.lib as pc
pp = pprint.PrettyPrinter()

HEAD_NODE_SUBNET = os.environ["HEAD_NODE_SUBNET"]
COMPUTE_NODE_SUBNET = os.environ["HEAD_NODE_SUBNET"]
KEY_NAME = os.environ["KEY_NAME"]
CONFIG = {'Image': {'Os': 'alinux2'},
          'HeadNode': {'InstanceType': 't2.large',
                       'Networking': {'SubnetId': HEAD_NODE_SUBNET},
                       'Ssh': {'KeyName': KEY_NAME}},
          'Scheduling': {'Scheduler': 'slurm',
                        'SlurmQueues':
```

```
[{'Name': 'queue0',
  'ComputeResources':
    [{'Name': 'queue0-i0', 'InstanceType': 't2.micro',
      'MinCount': 0, 'MaxCount': 10}],
  'Networking': {'SubnetIds': [COMPUTE_NODE_SUBNET]}]}
```

```
pp.pprint(pc.create_cluster(cluster_name="mycluster", cluster_configuration=CONFIG))
```

Salida:

```
{'cluster': {'cloudformationStackArn': 'arn:aws:cloudformation:us-
east-2:123456789012:stack/mycluster/00000000-aaaa-1111-999-000000000000',
  'cloudformationStackStatus': 'CREATE_IN_PROGRESS',
  'clusterName': 'mycluster',
  'clusterStatus': 'CREATE_IN_PROGRESS',
  'region': 'us-east-2',
  'scheduler': {'type': 'slurm'},
  'version': '3.7.0'}}
```

AWS Lambda para la biblioteca Python de AWS ParallelCluster

Puede implementar una capa de Lambda y un tiempo de ejecución para acceder a la biblioteca Python de AWS ParallelCluster. Alojamos archivos zip de AWS ParallelCluster que puede utilizar introduciendo el enlace al archivo zip tal y como se describe en los pasos siguientes. Lambda usa los archivos zip para preparar el entorno de tiempo de ejecución y permitir el acceso a la biblioteca Python. La biblioteca Python de AWS ParallelCluster se añade con AWS ParallelCluster versión 3.5.0. Solo puede usar la biblioteca para las versiones 3.5.0 y posteriores.

La URL del archivo zip alojado tiene el siguiente formato: `s3://aws-region-id-aws-parallelcluster/parallelcluster/3.7.0/layers/aws-parallelcluster/lambda-layer.zip`

Introducción al acceso a la biblioteca Python de AWS ParallelCluster con AWS Lambda

Creación de una capa de Lambda

1. Inicie sesión en AWS Management Console y vaya a la consola AWS Lambda.
2. En el panel de navegación, seleccione Capas y luego elija Crear capa.

3. Introduzca un nombre para la capa y seleccione Cargar un archivo desde Amazon S3.
4. Introduzca la URL del archivo zip: s3://*aws-region-id*-aws-parallelcluster/parallelcluster/*3.7.0*/layers/aws-parallelcluster/lambda-layer.zip.
5. En Arquitecturas compatibles, elija la arquitectura x86_64.
6. En Tiempos de ejecución compatibles, elija el tiempo de ejecución Python 3.9.
7. Seleccione Crear.

Uso de la capa de Lambda

1. En el panel de navegación de la consola Lambda, seleccione Funciones y, a continuación, Crear función.
2. Escriba un nombre para la función.
3. En Tiempo de ejecución, elija el tiempo de ejecución Python 3.9.
4. En Arquitectura, elija la arquitectura x86_64.
5. Elija Crear función.
6. Una vez creada la función, elija Capas y, a continuación, seleccione Añadir una capa.
7. Seleccione Capas personalizadas y elija la capa que creó en los pasos anteriores.
8. Elija la versión de la capa.
9. Elija Add (Agregar).
10. Su Lambda necesita permisos para administrar los clústeres creados con AWS ParallelCluster. Cree un rol de Lambda con los permisos que se indican en [Política de usuario básica pcluster de AWS ParallelCluster](#).

Ahora puede acceder a AWS ParallelCluster desde la biblioteca Python tal y como se describe en [API de la biblioteca Python de AWS ParallelCluster](#).

Cómo funciona AWS ParallelCluster

AWS ParallelCluster se creó no solo como una forma de administrar clústeres, sino también como referencia sobre cómo usar los servicios de AWS para crear el entorno de HPC.

Temas

- [AWS ParallelCluster procesos](#)
- [Servicios de AWS utilizados por AWS ParallelCluster](#)
- [AWS ParallelCluster Directorios internos](#)

AWS ParallelCluster procesos

Esta sección se aplica a los clústeres que se implementan con Slurm. Cuando se utiliza con este programador, AWS ParallelCluster gestiona el aprovisionamiento y la eliminación de los nodos de cómputo interactuando con el programador de tareas subyacente.

En el caso de los clústeres de HPC basados en AWS Batch, AWS ParallelCluster se basa en las capacidades que proporciona AWS Batch para la gestión de los nodos de procesamiento.

clustermgtd

Las siguientes tareas las realiza el daemon de administración de clústeres.

- Limpieza de particiones inactivas
- Gestión de las reservas de Slurm y los nodos asociados a los bloques de capacidad (consulte la siguiente sección)
- Administración de la capacidad estática: asegúrese de que la capacidad estática esté siempre activa y en buen estado
- Programador de sincronización con Amazon EC2.
- Limpieza de instancias huérfanas
- Restaure el estado del nodo del programador en la terminación de Amazon EC2 que se produce fuera del flujo de trabajo suspendido
- Administración de instancias de Amazon EC2 en mal estado (comprobaciones de estado de Amazon EC2 fallidas)

- Administración de eventos de mantenimiento programados
- Administración de los nodos del programador en mal estado (comprobaciones de estado del programador fallidas)

Gestión de las reservas de Slurm y los nodos asociados a los bloques de capacidad

ParallelCluster admite reservas de capacidad bajo demanda (ODCR) y bloques de capacidad para Machine Learning (CB). A diferencia de la ODCR, CB puede tener una hora de inicio futura y tiene un límite de tiempo.

Clustermgtd busca nodos en mal estado en un bucle y cierra cualquier instancia de EC2 que esté inactiva, sustituyéndola por instancias nuevas si son nodos estáticos.

ParallelCluster gestiona los nodos estáticos asociados a los bloques de capacidad de forma diferente. AWS ParallelCluster crea un clúster incluso si el CB aún no está activo, y las instancias se lanzan automáticamente una vez que el CB está activo.

Los nodos Slurm correspondientes a los recursos informáticos asociados a los CB que aún no están activos se mantienen en mantenimiento hasta que se alcanza la hora de inicio del CB. Los nodos de Slurm permanecerán en un estado de reserva o mantenimiento asociado al usuario administrador de Slurm, lo que significa que pueden aceptar trabajos, pero los trabajos permanecerán pendientes hasta que se elimine la reserva de Slurm.

Clustermgtd creará o eliminará automáticamente las reservas de Slurm y pondrá en mantenimiento los nodos CB relacionados en función del estado del CB. Cuando CB esté activo, se eliminará la reserva de Slurm, se iniciarán los nodos y estarán disponibles para los trabajos pendientes o para la presentación de nuevos trabajos.

Cuando se alcance la hora de finalización del CB, los nodos volverán a un estado de reserva o mantenimiento. Corresponde a los usuarios volver a enviar los trabajos o volver a ponerlos en cola en una nueva cola o recurso informático cuando el CB ya no esté activo y las instancias se cierren.

clusterstatusmgtd

El daemon de administración del estado del clúster administra la actualización del estado de la flota de computación. Cada minuto recupera el estado de la flota almacenado en una tabla de DynamoDB y administra cualquier solicitud STOP/START.

computemgtd

Los procesos de daemon (computemgtd) de administración de computación se ejecutan en cada uno de los nodos de computación del clúster. Cada cinco (5) minutos, el daemon de administración de computación confirma que se puede acceder al nodo principal y que está en buen estado. Si transcurren cinco (5) minutos durante los cuales no se puede acceder al nodo principal o este no está en buen estado, el nodo de computación se cierra.

Servicios de AWS utilizados por AWS ParallelCluster

Los siguientes servicios de Amazon Web Services (AWS) los utiliza AWS ParallelCluster.

Temas

- [Amazon API Gateway](#)
- [AWS Batch](#)
- [AWS CloudFormation](#)
- [Amazon CloudWatch](#)
- [CloudWatch Eventos de Amazon](#)
- [Amazon CloudWatch Logs](#)
- [AWS CodeBuild](#)
- [Amazon DynamoDB](#)
- [Amazon Elastic Block Store](#)
- [Amazon Elastic Compute Cloud](#)
- [Amazon Elastic Container Registry](#)
- [Amazon EFS](#)
- [Amazon FSx para Lustre](#)
- [Amazon FSx para ONTAP NetApp](#)
- [Amazon FSx para OpenZFS](#)
- [AWS Identity and Access Management](#)
- [AWS Lambda](#)
- [Amazon RDS](#)
- [Amazon Route 53](#)
- [Amazon Simple Notification Service](#)

- [Amazon Simple Storage Service](#)
- [Amazon VPC](#)
- [Elastic Fabric Adapter](#)
- [EC2 Image Builder](#)
- [NICE DCV](#)

Amazon API Gateway

Amazon API Gateway es un AWS servicio para crear, publicar, mantener, supervisar y proteger REST, HTTP y WebSocket API a cualquier escala

AWS ParallelCluster utiliza API Gateway para alojar la API de AWS ParallelCluster.

Para obtener más información sobre AWS Batch, consulte <https://aws.amazon.com/api-gateway/> y <https://docs.aws.amazon.com/apigateway/>.

AWS Batch

AWS Batch es un servicio del programador de trabajos administrado de AWS. Aprovisiona de forma dinámica la cantidad óptima y el tipo de recursos de computación (por ejemplo, instancias optimizadas para memoria o CPU) en los clústeres de AWS Batch. Estos recursos se aprovisionan en función de los requisitos específicos de sus trabajos por lotes, incluidos los requisitos de volumen. Con AWS Batch, no necesita instalar ni administrar clústeres de servidores o de software de computación por lotes adicionales para ejecutar sus trabajos de forma eficaz.

AWS Batch se usa solo con clústeres de AWS Batch.

Para obtener más información sobre AWS Batch, consulte <https://aws.amazon.com/batch/> y <https://docs.aws.amazon.com/batch/>.

AWS CloudFormation

AWS CloudFormation es un infrastructure-as-code servicio que proporciona un lenguaje común para modelar AWS y aprovisionar recursos de aplicaciones de terceros en su entorno de nube. Es el servicio principal utilizado por AWS ParallelCluster. Cada clúster de AWS ParallelCluster se representa como una pila y todos los recursos que necesita se definen en la plantilla AWS CloudFormation de AWS ParallelCluster. En la mayoría de los casos, los comandos de la CLI de AWS ParallelCluster se corresponden directamente con los comandos de la pila de AWS

CloudFormation, como los comandos de creación, actualización y eliminación. Las instancias que se lanzan dentro de un clúster realizan llamadas HTTPS al punto de conexión de AWS CloudFormation en la Región de AWS en la que se ha lanzado el clúster.

Para obtener más información sobre AWS CloudFormation, consulte <https://aws.amazon.com/cloudformation/> y <https://docs.aws.amazon.com/cloudformation/>.

Amazon CloudWatch

Amazon CloudWatch (CloudWatch) es un servicio de monitoreo y observabilidad que le proporciona datos e información procesable. Esta información se puede utilizar para supervisar las aplicaciones, responder a los cambios en el rendimiento y a las excepciones del servicio y optimizar la utilización de los recursos. EnAWS ParallelCluster, CloudWatch se usa como panel de control, para monitorear y registrar los pasos de creación de imágenes de Docker y el resultado de los trabajos. AWS Batch

Antes de AWS ParallelCluster la versión 2.10.0, solo CloudWatch se utilizaba con clústeres. AWS Batch

[Para obtener más información al respecto CloudWatch, consulte https://aws.amazon.com/cloudwatch/](https://aws.amazon.com/cloudwatch/) y <https://docs.aws.amazon.com/cloudwatch/>.

CloudWatch Eventos de Amazon

Amazon CloudWatch Events (CloudWatch Events) ofrece una transmisión casi en tiempo real de los eventos del sistema que describen los cambios en los recursos de Amazon Web Services (AWS). Mediante reglas sencillas que puede configurar rápidamente, puede asignar los eventos y dirigirlos a uno o más flujos o funciones de destino. EnAWS ParallelCluster, CloudWatch Events se usa para AWS Batch trabajos.

Para obtener más información sobre CloudWatch los eventos, consulte <https://docs.aws.amazon.com/eventbridge/latest/userguide/eb-cwe-now-eb>.

Amazon CloudWatch Logs

Amazon CloudWatch Logs (CloudWatch Logs) es una de las funciones principales de Amazon CloudWatch. Puede usarla para monitorear, almacenar, ver y buscar en los archivos de registro muchos de los componentes utilizados por AWS ParallelCluster.

Antes de AWS ParallelCluster la versión 2.6.0, CloudWatch Logs solo se utilizaba con AWS Batch clústeres.

Para más información, consulte [Integración con Amazon CloudWatch Logs](#).

AWS CodeBuild

AWS CodeBuild(CodeBuild) es un servicio de integración continua AWS gestionado que cumple con el código fuente, ejecuta pruebas y produce paquetes de software listos para su implementación. EnAWS ParallelCluster, CodeBuild se utiliza para crear imágenes de Docker de forma automática y transparente cuando se crean clústeres.

CodeBuild se usa solo con AWS Batch clústeres.

Para obtener más información al respecto CodeBuild, consulte <https://aws.amazon.com/codebuild/> y <https://docs.aws.amazon.com/codebuild/>.

Amazon DynamoDB

Amazon DynamoDB (DynamoDB) es un servicio de base de datos NoSQL rápido y flexible. Se utiliza para almacenar la información del estado mínimo del clúster. El nodo principal rastrea las instancias aprovisionadas en una tabla de DynamoDB.

DynamoDB no se usa con clústeres de AWS Batch.

Para obtener más información sobre DynamoDB, consulte <https://aws.amazon.com/dynamodb/> y <https://docs.aws.amazon.com/dynamodb/>.

Amazon Elastic Block Store

Amazon Elastic Block Store (Amazon EBS) es un servicio de almacenamiento en bloque de alto rendimiento que proporciona almacenamiento persistente para volúmenes compartidos. Todos los ajustes de Amazon EBS se pueden transferir a través de la configuración. Los volúmenes de Amazon EBS se pueden inicializar vacíos o a partir de una instantánea de Amazon EBS existente.

Para obtener más información sobre Amazon EBS, consulte <https://aws.amazon.com/ebs/> y <https://docs.aws.amazon.com/ebs/>.

Amazon Elastic Compute Cloud

Amazon Elastic Compute Cloud (Amazon EC2) proporciona la capacidad de computación para AWS ParallelCluster. Los nodos principal y de computación son instancias de Amazon EC2. Se puede

seleccionar cualquier tipo de instancia compatible con HVM. Los nodos principal y de computación pueden ser de distintos tipos de instancias. Además, si se utilizan varias colas, algunos o todos los nodos de computación también se pueden lanzar como una instancia de spot. Los volúmenes de almacén de instancias que se encuentran en las instancias se montan como volúmenes LVM fragmentados.

Para obtener más información sobre Amazon EC2, consulte <https://aws.amazon.com/ec2/> y <https://docs.aws.amazon.com/ec2/>.

Amazon Elastic Container Registry

Amazon Elastic Container Registry (Amazon ECR) es un registro de contenedores de Docker completamente administrado que facilita el almacenamiento, la administración y la implementación de imágenes de contenedores de Docker. En AWS ParallelCluster, Amazon ECR almacena las imágenes de Docker que se crean al crear los clústeres. Luego, AWS Batch utiliza dichas imágenes para ejecutar los contenedores para los trabajos enviados.

Amazon ECR se usa solo con clústeres de AWS Batch.

Para obtener más información, consulte <https://aws.amazon.com/ecr/> y <https://docs.aws.amazon.com/ecr/>.

Amazon EFS

Amazon Elastic File System (Amazon EFS) proporciona un sistema de archivos NFS elástico completamente administrado que se utiliza con servicios de Nube de AWS y recursos en las instalaciones. Amazon EFS se utiliza cuando se especifica [EfsSettings](#). Se ha agregado la compatibilidad de Amazon EFS en AWS ParallelCluster versión 2.1.0.

Para obtener más información sobre Amazon EFS, consulte <https://aws.amazon.com/efs/> y <https://docs.aws.amazon.com/efs/>.

Amazon FSx para Lustre

FSx for Lustre proporciona un sistema de archivos de alto rendimiento que utiliza el sistema de archivos Lustre de código abierto. FSx for Lustre se utiliza cuando se especifica [Propiedades de FsxLustreSettings](#). Se ha agregado la compatibilidad de FSx for Lustre en AWS ParallelCluster versión 2.2.1.

Para obtener más información sobre FSx for Lustre, consulte <https://aws.amazon.com/fsx/lustre/> y <https://docs.aws.amazon.com/fsx/>.

Amazon FSx para ONTAP NetApp

FSx for ONTAP proporciona un sistema de almacenamiento compartido totalmente gestionado basado en el popular sistema NetApp de archivos ONTAP. FSx for ONTAP se utiliza cuando se especifica [Propiedades de FsxOntapSettings](#). Se ha agregado la compatibilidad de FSx for ONTAP en AWS ParallelCluster versión 3.2.0.

Para obtener más información sobre FSx for ONTAP, consulte <https://aws.amazon.com/fsx/netapp-ontap/> y <https://docs.aws.amazon.com/fsx/>.

Amazon FSx para OpenZFS

FSx for OpenZFS proporciona un sistema de almacenamiento compartido totalmente administrado basado en el popular sistema de archivos OpenZFS. FSx for OpenZFS se utiliza cuando se especifica [Propiedades de FsxOpenZfsSettings](#). Se ha agregado la compatibilidad de FSx for OpenZFS en AWS ParallelCluster versión 3.2.0.

Para obtener más información sobre FSx for OpenZFS, consulte <https://aws.amazon.com/fsx/openzfs/> y <https://docs.aws.amazon.com/fsx/>.

AWS Identity and Access Management

AWS Identity and Access Management (IAM) se utiliza en AWS ParallelCluster para proporcionar un rol de IAM con menos privilegios para Amazon EC2 para la instancia que es específica de cada clúster individual. A las instancias de AWS ParallelCluster se les da acceso solo a las llamadas a la API específicas necesarias para implementar y administrar el clúster.

Con los clústeres de AWS Batch, también se crean roles de IAM para los componentes que se incluyen en el proceso de creación de la imagen de Docker cuando se crean los clústeres. Estos componentes incluyen las funciones de Lambda que permiten añadir y eliminar imágenes de Docker desde y hacia el repositorio de Amazon ECR. También incluyen las funciones que permiten eliminar el bucket de Amazon S3 que se creó para el clúster y el CodeBuild proyecto. También hay roles para los recursos, las instancias y los trabajos de AWS Batch.

Para obtener más información sobre IAM, consulte <https://aws.amazon.com/iam/> y <https://docs.aws.amazon.com/iam/>.

AWS Lambda

AWS Lambda (Lambda) ejecuta las funciones que organizan la creación de imágenes de Docker. Lambda también se encarga de la limpieza de recursos del clúster personalizado, como las imágenes de Docker almacenadas en el repositorio de Amazon ECR y en Amazon S3.

Para obtener más información sobre Lambda, consulte <https://aws.amazon.com/lambda/> y <https://docs.aws.amazon.com/lambda/>.

Amazon RDS

Amazon Relational Database Service (Amazon RDS) es un servicio web que facilita la configuración, el funcionamiento y el escalado de una base de datos relacional en la nube de AWS.

AWS ParallelCluster utiliza Amazon RDS para AWS Batch y Slurm.

Para obtener más información sobre Amazon RDS, consulte <https://aws.amazon.com/rds/> y <https://docs.aws.amazon.com/rds/>.

Amazon Route 53

Amazon Route 53 (Route 53) se utiliza para crear zonas alojadas con nombres de host y nombres de dominio totalmente cualificados para cada uno de los nodos de computación.

Para obtener más información sobre Route 53, consulte <https://aws.amazon.com/route53/> y <https://docs.aws.amazon.com/route53/>.

Amazon Simple Notification Service

(Amazon SNS) es un servicio administrado que brinda la entrega de mensajes de los publicadores a los suscriptores (también conocidos como productores y consumidores).

AWS ParallelCluster utiliza Amazon SNS para el alojamiento de la API.

Para obtener más información sobre Amazon SNS, consulte <https://aws.amazon.com/sns/> y <https://docs.aws.amazon.com/sns/>.

Amazon Simple Storage Service

Amazon Simple Storage Service (Amazon S3) almacena plantillas de AWS ParallelCluster ubicadas en cada Región de AWS. AWS ParallelCluster se puede configurar para permitir que las herramientas de CLI/SDK utilicen Amazon S3.

AWS ParallelCluster también crea un bucket de Amazon S3 en su Cuenta de AWS para almacenar los recursos que utilizan los clústeres, como el archivo de configuración del clúster. AWS ParallelCluster mantiene un bucket de Amazon S3 en cada Región de AWS en la que cree clústeres.

Cuando utiliza un clúster de AWS Batch, se utiliza un bucket de Amazon S3 en su cuenta para almacenar los datos relacionados. Por ejemplo, el bucket almacena artefactos creados cuando se crea una imagen de Docker y scripts de los trabajos enviados.

Para obtener más información, consulte <https://aws.amazon.com/s3/> y <https://docs.aws.amazon.com/s3/>.

Amazon VPC

Una Amazon VPC define una red utilizada por los nodos del clúster.

Para obtener más información sobre Amazon VPC, consulte <https://aws.amazon.com/vpc/> y <https://docs.aws.amazon.com/vpc/>.

Elastic Fabric Adapter

Elastic Fabric Adapter (EFA) es una interfaz de red para instancias de Amazon EC2 que pueden usar los clientes para ejecutar aplicaciones que requieren altos niveles de comunicaciones entre nodos a escala en AWS.

Para obtener más información sobre EC2 Image Builder, consulte <https://aws.amazon.com/hpc/efa/>.

EC2 Image Builder

EC2 Image Builder es un servicio totalmente AWS gestionado que le ayuda a automatizar la creación, la administración y el despliegue de imágenes personalizadas, seguras up-to-date y de servidor.

AWS ParallelCluster usa Image Builder para crear y administrar imágenes de AWS ParallelCluster.

Para obtener más información sobre EC2 Image Builder, consulte <https://aws.amazon.com/image-builder/> y <https://docs.aws.amazon.com/imagebuilder/>.

NICE DCV

NICE DCV es un protocolo de pantalla remota de alto rendimiento que proporciona una forma segura de entregar escritorios remotos y transmisión de aplicaciones a cualquier dispositivo en

diferentes condiciones de red. Se utiliza NICE DCV cuando se especifica la configuración [Sección de HeadNode/Dcv](#). Se ha agregado la compatibilidad de NICE DCV en AWS ParallelCluster versión 2.5.0.

Para obtener más información sobre NICE DCV, consulte <https://aws.amazon.com/hpc/dcv/> y <https://docs.aws.amazon.com/dcv/>.

AWS ParallelCluster Directorios internos

Hay varios directorios internos que se AWS ParallelCluster utilizan para compartir datos dentro del clúster. Los siguientes directorios se comparten entre el nodo principal, los nodos de procesamiento y los nodos de inicio de sesión:

`/opt/slurm`

`/opt/intel`

`/opt/parallelcluster/shared` (only with compute nodes)

`/opt/parallelcluster/shared_login_nodes` (only with login nodes)

`/home` (unless specified in SharedStorage)

Note

De forma predeterminada, estos directorios se crean en el volumen EBS del nodo principal y se comparten a medida que NFS exporta a los nodos de procesamiento e inicio de sesión. A partir de la versión AWS ParallelCluster 3.8, puede AWS ParallelCluster habilitar la creación y administración de un sistema de archivos Amazon EFS para alojar y compartir estos directorios configurando el parámetro en `efs`. [SharedStorageType](#)

Cuando el clúster se amplía, las exportaciones de NFS a través del volumen de EBS pueden provocar cuellos de botella en el rendimiento. Con EFS, puede evitar las exportaciones de NFS a medida que el clúster se amplía y evitar los cuellos de botella de rendimiento asociados a ellas.

Tutoriales

En los siguientes tutoriales se le muestra cómo comenzar a utilizar AWS ParallelCluster versión 3 y se proporciona orientación sobre prácticas recomendadas para algunas tareas comunes.

Al utilizar la interfaz de línea de comandos (CLI) o API de AWS ParallelCluster, solo paga por los recursos de AWS que se crean al crear o actualizar imágenes y clústeres de AWS ParallelCluster. Para más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

La interfaz de usuario de AWS ParallelCluster se basa en una arquitectura sin servidor y, en la mayoría de los casos, se puede utilizar en la categoría de nivel AWS gratuito. Para obtener más información, consulte [Costes de la interfaz de usuario de AWS ParallelCluster](#).

Temas

- [Ejecución del primer trabajo en AWS ParallelCluster](#)
- [Creación de una AWS ParallelCluster AMI personalizada](#)
- [Integración de Active Directory](#)
- [Configuración del cifrado de almacenamiento compartido con una clave AWS KMS](#)
- [Ejecución de trabajos en un clúster en modo de cola múltiple](#)
- [Uso de la API AWS ParallelCluster](#)
- [Crear un clúster con contabilidad Slurm](#)
- [Volver a una versión anterior del documento de Systems Manager de AWS](#)
- [Crear un clúster con AWS CloudFormation](#)
- [AWS ParallelClusterIntegración de la interfaz de usuario con Identity Center](#)

Ejecución del primer trabajo en AWS ParallelCluster

Este tutorial le guía por la ejecución de su primer trabajo Hello World en AWS ParallelCluster

Al utilizar la interfaz de línea de comandos (CLI) o API de AWS ParallelCluster, solo paga por los recursos de AWS que se crean al crear o actualizar imágenes y clústeres de AWS ParallelCluster. Para obtener más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

La interfaz de usuario de AWS ParallelCluster se basa en una arquitectura sin servidor y, en la mayoría de los casos, se puede utilizar en la categoría de nivel AWS gratuito. Para obtener más información, consulte [Costes de la interfaz de usuario de AWS ParallelCluster](#).

Requisitos previos

- AWS ParallelCluster [está instalado](#).
- AWS CLI está [instalado y configurado](#).
- Tiene un [par de claves EC2](#).
- Tiene un rol de IAM con los [permisos](#) necesarios para ejecutar el CLI [pcluster](#).

Comprobación de la instalación

En primer lugar, comprobamos que AWS ParallelCluster esté instalado y configurado correctamente, incluyendo la dependencia Node.js.

```
$ node --version
v16.8.0
$ pcluster version
{
  "version": "3.7.0"
}
```

Esto devuelve la versión en ejecución de AWS ParallelCluster.

Creación de su primer clúster

Ahora ha llegado el momento de crear su primer clúster. Debido a que la carga de trabajo de este tutorial no es excesiva, podemos usar el tamaño de instancia predeterminado de `t2.micro`. (Para las cargas de trabajo de producción, puede elegir el tamaño de instancia que mejor se adapte a sus necesidades). Llamemos a su clúster `hello-world`.

```
$ pcluster create-cluster \
  --cluster-name hello-world \
  --cluster-configuration hello-world.yaml
```

Note

Debe especificarse el Región de AWS a especificar para la mayoría de `pcluster` los comandos. Si no se especifica en la variable de entorno `AWS_DEFAULT_REGION`, o el ajuste `region` en la sección `[default]` del archivo `~/.aws/config`, el parámetro `--region` debe proporcionarse en la línea de comando `pcluster`.

Si recibe un mensaje del resultado sobre la configuración, tiene que ejecutar lo siguiente para configurar :

```
$ pcluster configure --config hello-world.yaml
```

Si el comando `pcluster create-cluster` se ejecuta correctamente, verá un resultado similar al siguiente:

```
{
  "cluster": {
    "clusterName": "hello-world",
    "cloudformationStackStatus": "CREATE_IN_PROGRESS",
    "cloudformationStackArn": "arn:aws:cloudformation:xxx:stack/xxx",
    "region": "...",
    "version": "...",
    "clusterStatus": "CREATE_IN_PROGRESS"
  }
}
```

La creación del clúster se supervisa mediante:

```
$ pcluster describe-cluster --cluster-name hello-world
```

Los `clusterStatus` informes "CREATE_IN_PROGRESS" mientras se crea el clúster. `clusterStatus` pasa a "CREATE_COMPLETE" cuando el clúster se haya creado correctamente. El resultado también nos proporciona las `publicIpAddress` y `privateIpAddress` de nuestro nodo principal.

Iniciar sesión en su nodo principal

Use su archivo OpenSSH para iniciar sesión en el nodo principal.

```
$ pcluster ssh --cluster-name hello-world -i /path/to/keyfile.pem
```

Después de iniciar sesión, ejecute el comando `sinfo` para comprobar que sus nodos de computación se instalan y configuran.

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
```

```
queue1*      up    infinite    10  idle~ queue1-dy-queue1t2micro-[1-10]
```

El resultado muestra que tenemos una cola en nuestro clúster, con hasta diez nodos.

Ejecución de su primer trabajo con Slurm

A continuación, creamos un trabajo que entra en suspensión durante un tiempo y luego genera su propio nombre de host. Cree un archivo denominado `hellojob.sh` con el siguiente contenido.

```
#!/bin/bash
sleep 30
echo "Hello World from $(hostname)"
```

A continuación, envíe el trabajo mediante `sbatch` y compruebe que se ejecuta.

```
$ sbatch hellojob.sh
Submitted batch job 2
```

Ahora puede ver la cola y comprobar el estado del trabajo. El aprovisionamiento de una nueva instancia Amazon EC2 se inicia en segundo plano. Puede monitorizar el estado de las instancias del clúster con el comando `sinfo`.

```
$ squeue
          JOBID PARTITION    NAME    USER ST      TIME  NODES NODELIST(REASON)
           2      queue1 hellojob ec2-user CF       3:30      1 queue1-dy-
queue1t2micro-1
```

El resultado muestra que se ha enviado el trabajo a `queue1`. Espere 30 segundos a que el trabajo se termine y, a continuación, vuelva a ejecutar `squeue`.

```
$ squeue
          JOBID PARTITION    NAME    USER ST      TIME  NODES NODELIST(REASON)
```

Ahora que no hay trabajos en la cola, podemos comprobar el resultado en nuestro directorio actual.

```
$ ls -l
total 8
-rw-rw-r-- 1 ec2-user ec2-user 57 Sep  1 14:25 hellojob.sh
-rw-rw-r-- 1 ec2-user ec2-user 43 Sep  1 14:30 slurm-2.out
```

En el resultado, vemos un archivo "out". Podemos ver el resultado de nuestro trabajo:

```
$ cat slurm-2.out
Hello World from queue1-dy-queue1t2micro-1
```

El resultado también muestra que nuestro trabajo se ha ejecutado correctamente en la instancia queue1-dy-queue1t2micro-1.

En el clúster que acaba de crear, solo el directorio principal se comparte entre todos los nodos del clúster.

Para obtener más información acerca de la creación y el uso de clústeres, consulte [Prácticas recomendadas](#).

Si la aplicación requiere software, bibliotecas o datos compartidos, considere las siguientes opciones:

- Cree una AMI personalizada habilitada AWS ParallelCluster que incluya su software, tal y como se describe en [Creación de una AWS ParallelCluster AMI personalizada](#).
- Utilice la opción [StorageSettings](#) del archivo de AWS ParallelCluster configuración para especificar un sistema de archivos compartido y almacenar el software instalado en la ubicación de montaje especificada.
- Use [Acciones de arranque personalizadas](#) para automatizar el procedimiento de arranque de cada nodo del clúster.

Creación de una AWS ParallelCluster AMI personalizada

Al utilizar la interfaz de línea de AWS ParallelCluster comandos (CLI) o la API, solo paga por los AWS recursos que se crean al crear o actualizar AWS ParallelCluster imágenes y clústeres. Para obtener más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

La AWS ParallelCluster interfaz de usuario se basa en una arquitectura sin servidor y, en la mayoría de los casos, se puede utilizar en la categoría de capa AWS gratuita. Para obtener más información, consulte [Costes de la interfaz de usuario de AWS ParallelCluster](#).

Important

Si crea una AMI personalizada, debe repetir los pasos que utilizó para crear la AMI personalizada con cada nueva versión de AWS ParallelCluster .

Antes de seguir leyendo le recomendamos revisar la sección [Acciones de arranque personalizadas](#). Determine si las modificaciones que desea realizar se pueden incluir en un script y son compatibles con futuras versiones de AWS ParallelCluster .

Si bien crear una AMI personalizada en general no es lo ideal, hay situaciones específicas en las que AWS ParallelCluster es necesario crear una AMI personalizada. En este tutorial encontrará una introducción a una AMI personalizada para estos escenarios.

Requisitos previos

- AWS ParallelCluster [está instalado](#).
- AWS CLI [está instalado y configurado](#).
- Tiene un [par de claves EC2](#).
- Tiene un rol de IAM con los [permisos](#) necesarios para ejecutar el CLI [pcluster](#) y crear imágenes.

Cómo personalizar la AWS ParallelCluster AMI

Hay dos formas de crear una AWS ParallelCluster AMI personalizada. Uno de estos dos métodos consiste en crear una AMI nueva mediante la AWS ParallelCluster CLI. Otro método requiere que realice modificaciones manuales para crear una nueva AMI que esté disponible en su Cuenta de AWS.

Cree una AWS ParallelCluster AMI personalizada

Si tiene una AMI y un software personalizados, puede aplicar los cambios necesarios además AWS ParallelCluster de ellos. AWS ParallelCluster se basa en el servicio EC2 Image Builder para crear AMI personalizadas. Para obtener más información, consulte la [Guía de usuario de Generador de imágenes](#).

Puntos clave

- El proceso tarda unos 60 minutos. Este tiempo puede variar si hay más [Build/Components](#) que instalar en el momento de la construcción.
- La AMI está etiquetada con las versiones de los componentes principales. Estas incluyen el núcleo, el programador y el controlador [EFA](#). En la descripción de la AMI también se indica un subconjunto de las versiones de los componentes.

- A partir de la AWS ParallelCluster versión 3.0.0, se puede utilizar un nuevo conjunto de comandos CLI para gestionar el ciclo de vida de las imágenes. Esto incluye [build-image](#), [list-images](#), [describe-image](#) y [delete-image](#).
- Este método se puede repetir. Puede volver a ejecutarlo para mantener las AMI actualizadas (por ejemplo, las actualizaciones del sistema operativo) y, a continuación, utilizarlas cuando actualice un clúster existente.

Note

Si utiliza este método en la partición de AWS China, es posible que se produzcan errores de red. Por ejemplo, es posible que veas estos errores en el `pcluster build-image` comando cuando descarga paquetes desde GitHub o desde un repositorio del sistema operativo. Si esto ocurre, le recomendamos que utilice uno de los siguientes métodos alternativos:

1. Siga el [Modificar una AWS ParallelCluster AMI](#) enfoque que omita este comando.
2. Cree la imagen en otra partición o región, por ejemplo, y, a continuación `us-east-1`, almacénela y restáurela para moverla a la región de China. Para obtener más información, consulte [Almacenamiento y restauración de una AMI mediante S3](#) en la Guía del usuario de Amazon EC2.

Pasos:

1. Configure sus Cuenta de AWS credenciales para que el AWS ParallelCluster cliente pueda realizar llamadas a las operaciones de la AWS API en su nombre. Para obtener una lista de los permisos necesarios, consulte [AWS Identity and Access Management permisos en AWS ParallelCluster](#).
2. Cree un archivo de configuración de imagen de compilación básico. Para ello, especifique el [InstanceType](#) que se utilizará para crear la imagen y el [ParentImage](#). Se utilizan como punto de partida para crear la AMI. Para obtener más información acerca de los parámetros de configuración opcionales, consulte [Configuración de imagen](#).

Build:

```
InstanceType: <BUILD_INSTANCE_TYPE>  
ParentImage: <BASE_AMI_ID>
```

3. Utilice el comando CLI `pcluster build-image` para crear una AWS ParallelCluster AMI a partir de la AMI que proporciona como base.

```
$ pcluster build-image --image-id IMAGE_ID --image-configuration IMAGE_CONFIG.yaml --  
region REGION  
  {  
    "image": {  
      "imageId": "IMAGE_ID",  
      "imageBuildStatus": "BUILD_IN_PROGRESS",  
      "cloudformationStackStatus": "CREATE_IN_PROGRESS",  
      "cloudformationStackArn": "arn:aws:cloudformation:us-east-1:123456789012:stack/  
IMAGE_ID/abcd1234-ef56-gh78-ij90-1234abcd5678",  
      "region": "us-east-1",  
      "version": "3.7.0"  
    }  
  }
```

Warning

`pcluster build-image` utiliza la VPC predeterminada. Si elimina la VPC predeterminada mediante AWS Control Tower una zona de AWS destino, el ID de subred debe especificarse en el archivo de configuración de la imagen. Para obtener más información, consulte [SubnetId](#).

Para obtener una lista de otros parámetros, consulte la página de referencia de comandos [pcluster build-image](#). Los resultados del comando anterior son los siguientes:

- Se crea una CloudFormation pila en función de la configuración de la imagen. La pila incluye todos los recursos de Generador de imágenes de EC2 necesarios para la compilación.
- Los recursos creados incluyen los AWS ParallelCluster componentes oficiales de Image Builder a los que se pueden añadir componentes personalizados de Image Builder. Para aprender a crear componentes personalizados, consulte los [ejemplos de AMI personalizadas](#) en el taller de HPC para clientes del sector público.
- EC2 Image Builder lanza una instancia de compilación, aplica AWS ParallelCluster el libro de cocina, instala la pila de software y realiza AWS ParallelCluster las tareas de configuración necesarias. El AWS ParallelCluster libro de cocina se utiliza para compilar y arrancar. AWS ParallelCluster
- La instancia se detiene y se crea una nueva AMI a partir de ella.

- Se lanza otra instancia desde la AMI recién creada. Durante la fase de prueba, Generador de imágenes de EC2 ejecuta las pruebas definidas en los componentes de Generador de imágenes.
 - Si la compilación se realiza correctamente, se elimina la pila. Si la compilación falla, la pila se retiene y está disponible para su inspección.
4. Puede monitorear el estado del proceso de compilación si ejecuta el siguiente comando. Una vez completada la compilación, puede ejecutarla para recuperar el ID de AMI indicado en la respuesta.

```
$ pcluster describe-image --image-id IMAGE_ID --region REGION

# BEFORE COMPLETE
{
  "imageConfiguration": {
    "url": "https://parallelcluster-1234abcd5678efgh-v1-do-not-
delete.s3.amazonaws.com/parallelcluster/3.7.0/images/IMAGE_ID-abcd1234efgh5678/
configs/image-config.yaml?...",
  },
  "imageId": "IMAGE_ID",
  "imagebuilderImageStatus": "BUILDING",
  "imageBuildStatus": "BUILD_IN_PROGRESS",
  "cloudformationStackStatus": "CREATE_IN_PROGRESS",
  "cloudformationStackArn": "arn:aws:cloudformation:us-east-1:123456789012:stack/
IMAGE_ID/abcd1234-ef56-gh78-ij90-1234abcd5678",
  "region": "us-east-1",
  "version": "3.7.0",
  "cloudformationStackTags": [
    {
      "value": "3.7.0",
      "key": "parallelcluster:version"
    },
    {
      "value": "IMAGE_ID",
      "key": "parallelcluster:image_name"
    },
    ...
  ],
  "imageBuildLogsArn": "arn:aws:logs:us-east-1:123456789012:log-group:/aws/
imagebuilder/ParallelClusterImage-IMAGE_ID",
  "cloudformationStackCreationTime": "2022-04-05T21:36:26.176Z"
}

# AFTER COMPLETE
```

```
{
  "imageConfiguration": {
    "url": "https://parallelcluster-1234abcd5678efgh-v1-do-not-delete.s3.us-east-1.amazonaws.com/parallelcluster/3.7.0/images/IMAGE_ID-abcd1234efgh5678/configs/image-config.yaml?Signature=..."
  },
  "imageId": "IMAGE_ID",
  "imageBuildStatus": "BUILD_COMPLETE",
  "region": "us-east-1",
  "ec2AmiInfo": {
    "amiName": "IMAGE_ID 2022-04-05T21-39-24.020Z",
    "amiId": "ami-1234stuv5678wxyz",
    "description": "AWS ParallelCluster AMI for alinux2, kernel-4.14.238-182.422.amzn2.x86_64, lustre-2.10.8-5.amzn2.x86_64, efa-1.13.0-1.amzn2.x86_64, dcv-2021.1.10598-1.el7.x86_64, slurm-20-11-8-1",
    "state": "AVAILABLE",
    "tags": [
      {
        "value": "2021.3.11591-1.el7.x86_64",
        "key": "parallelcluster:dcv_version"
      },
      ...
    ],
    "architecture": "x86_64"
  },
  "version": "3.7.0"
}
```

5. Para crear el clúster, especifique el ID de AMI en el campo [CustomAmi](#) de la configuración del clúster.

Solución de problemas y supervisión del proceso de creación de AMI

La creación de la imagen se completa en aproximadamente una hora. Puede supervisar el proceso ejecutando el comando [pcluster describe-image](#) o los comandos de recuperación de registros.

```
$ pcluster describe-image --image-id IMAGE_ID --region REGION
```

El [build-image](#) comando crea una CloudFormation pila con todos los recursos de EC2 necesarios para crear la imagen e inicia el proceso de EC2 Image Builder.

Tras ejecutar el [build-image](#) comando, es posible recuperar los eventos de la CloudFormation pila mediante. [pcluster get-image-stack-events](#) Puede filtrar los resultados con el `--query` parámetro para ver los eventos más recientes. Para obtener más información, consulte [Filtrar los AWS CLI resultados](#) en la Guía del AWS Command Line Interface usuario.

```
$ pcluster get-image-stack-events --image-id IMAGE_ID --region REGION --query
"events[0]"
{
  "eventId": "ParallelClusterImage-CREATE_IN_PROGRESS-2022-04-05T21:39:24.725Z",
  "physicalResourceId": "arn:aws:imagebuilder:us-east-1:123456789012:image/
parallelclusterimage-IMAGE_ID/3.7.0/1",
  "resourceStatus": "CREATE_IN_PROGRESS",
  "resourceStatusReason": "Resource creation Initiated",
  "resourceProperties": "{\"InfrastructureConfigurationArn\":
\\\"arn:aws:imagebuilder:us-east-1:123456789012:infrastructure-configuration/
parallelclusterimage-abcd1234-ef56-gh78-ij90-1234abcd5678\\\",\\\"ImageRecipeArn\\\":
\\\"arn:aws:imagebuilder:us-east-1:123456789012:image-recipe/parallelclusterimage-
IMAGE_ID/3.7.0\\\",\\\"DistributionConfigurationArn\\\":\\\"arn:aws:imagebuilder:us-
east-1:123456789012:distribution-configuration/parallelclusterimage-abcd1234-ef56-
gh78-ij90-1234abcd5678\\\",\\\"Tags\\\":{\\\"parallelcluster:image_name\\\":\\\"IMAGE_ID\\\",
\\\"parallelcluster:image_id\\\":\\\"IMAGE_ID\\\"}}\",
  "stackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/IMAGE_ID/abcd1234-
ef56-gh78-ij90-1234abcd5678",
  "stackName": "IMAGE_ID",
  "logicalResourceId": "ParallelClusterImage",
  "resourceType": "AWS::ImageBuilder::Image",
  "timestamp": "2022-04-05T21:39:24.725Z"
}
```

Después de unos 15 minutos, los eventos de la pila aparecen en la entrada del evento de registro relacionada con la creación de Generador de imágenes. Ahora puede enumerar los flujos de registro de imágenes y supervisar los pasos de Generador de imágenes mediante los comandos [pcluster list-image-log-streams](#) y [pcluster get-image-log-events](#).

```
$ pcluster list-image-log-streams --image-id IMAGE_ID --region REGION \
--query 'logStreams[*].logStreamName'

"3.7.0/1"
]

$ pcluster get-image-log-events --image-id IMAGE_ID --region REGION \
--log-stream-name 3.7.0/1 --limit 3
```

```
{
  "nextToken": "f/36295977202298886557255241372854078762600452615936671762",
  "prevToken": "b/36295977196879805474012299949460899222346900769983430672",
  "events": [
    {
      "message": "ExecuteBash: FINISHED EXECUTION",
      "timestamp": "2022-04-05T22:13:26.633Z"
    },
    {
      "message": "Document arn:aws:imagebuilder:us-east-1:123456789012:component/parallelclusterimage-test-abcd1234-ef56-gh78-ij90-1234abcd5678/3.7.0/1",
      "timestamp": "2022-04-05T22:13:26.741Z"
    },
    {
      "message": "TOE has completed execution successfully",
      "timestamp": "2022-04-05T22:13:26.819Z"
    }
  ]
}
```

Continúe comprobando con el comando [describe-image](#) hasta que vea el estado BUILD_COMPLETE.

```
$ pcluster describe-image --image-id IMAGE_ID --region REGION
{
  "imageConfiguration": {
    "url": "https://parallelcluster-1234abcd5678efgh-v1-do-not-delete.s3.us-east-1.amazonaws.com/parallelcluster/3.7.0/images/IMAGE_ID-abcd1234efgh5678/configs/image-config.yaml?Signature=..."
  },
  "imageId": "IMAGE_ID",
  "imageBuildStatus": "BUILD_COMPLETE",
  "region": "us-east-1",
  "ec2AmiInfo": {
    "amiName": "IMAGE_ID 2022-04-05T21-39-24.020Z",
    "amiId": "ami-1234stuv5678wxyz",
    "description": "AWS ParallelCluster AMI for alinux2, kernel-4.14.238-182.422.amzn2.x86_64, lustre-2.10.8-5.amzn2.x86_64, efa-1.13.0-1.amzn2.x86_64, dcw-2021.1.10598-1.el7.x86_64, slurm-20-11-8-1",
    "state": "AVAILABLE",
    "tags": [
      {
        "value": "2021.3.11591-1.el7.x86_64",
```

```

    "key": "parallelcluster:dcv_version"
  },
  ...
],
"architecture": "x86_64"
},
"version": "3.7.0"
}

```

Si necesita solucionar un problema de creación de una AMI personalizada, cree un archivo con los registros de imágenes tal y como se describe en los pasos siguientes.

Puede archivar los registros en Amazon S3 o en un archivo local (según el parámetro `--output`).

```

$ pcluster export-image-logs --image-id IMAGE_ID --region REGION \
--bucket BUCKET_NAME --bucket-prefix BUCKET_FOLDER
{
  "url": "https://BUCKET_NAME.s3.us-east-1.amazonaws.com/BUCKET-FOLDER/IMAGE_ID-
logs-202209071136.tar.gz?AWSAccessKeyId=..."
}

$ pcluster export-image-logs --image-id IMAGE_ID \
--region REGION --bucket BUCKET_NAME --bucket-prefix BUCKET_FOLDER --output-file /tmp/
archive.tar.gz
{
  "path": "/tmp/archive.tar.gz"
}

```

El archivo contiene los flujos de CloudWatch registros relacionados con el proceso de Image Builder y los eventos de AWS CloudFormation pila. El proceso puede tardar varios minutos en completarse.

Administración de AMI personalizadas

A partir de la AWS ParallelCluster versión 3.0.0, se ha agregado un nuevo conjunto de comandos a la CLI para crear, monitorear y administrar el ciclo de vida de la imagen. Para obtener más información sobre el comando, consulte [Comandos pcluster](#).

Modificar una AWS ParallelCluster AMI

Este método consiste en modificar una AWS ParallelCluster AMI oficial añadiéndole personalización por encima. Las AWS ParallelCluster AMI base se actualizan con las nuevas versiones. Estas AMI

tienen todos los componentes necesarios AWS ParallelCluster para funcionar cuando se instalan y configuran. Puede empezar con una de ellas como base.

Puntos clave:

- Este método es más rápido que el comando [build-image](#). Sin embargo, es un proceso manual y no se puede repetir automáticamente.
- Con este método, no tiene acceso a los comandos de recuperación de registros y administración del ciclo de vida de las imágenes que están disponibles a través de la CLI.

Pasos:

New EC2 console

1. Busque la AMI que corresponda a la específica Región de AWS que utiliza. Para encontrarlo, utilice el [pcluster list-official-images](#) comando con el `--region` parámetro para seleccionar los `--architecture` parámetros Región de AWS y `--os` y filtros específicos para la AMI deseada con el sistema operativo y la arquitectura que desee utilizar. De la salida, recupere el ID de imagen del EC2.
2. [Inicie sesión en la consola Amazon EC2 AWS Management Console y ábrala en https://console.aws.amazon.com/ec2/](https://console.aws.amazon.com/ec2/).
3. En el panel de navegación, elija Imágenes y a continuación AMI. Busque el ID de imagen EC2 recuperado, seleccione la AMI y elija Lanzar instancia a partir de una AMI.
4. Desplácese hacia abajo y elija el tipo de instancia.
5. Elija su Par de claves y Lanzar instancias.
6. Inicie sesión en la instancia mediante el usuario del sistema operativo y su clave de SSH.
7. Personalice la instancia de forma manual para cumplir con sus requisitos.
8. Ejecute el siguiente comando para preparar su instancia para la creación de la AMI:

```
sudo /usr/local/sbin/ami_cleanup.sh
```

9. Desde la consola, seleccione Estado de instancia y a continuación seleccione Detener instancia.

Navegue a Instancias, elija la nueva instancia, seleccione Estado de la instancia, y Detener instancia

10. Cree una AMI nueva a partir de la instancia mediante la consola EC2 o AWS CLI [cree](#) una imagen.

Desde la consola EC2

- a. Elija Instances (Instancia[s]) en el panel de navegación.
- b. Elija la instancia que creó y modificó.
- c. Elija Acciones, Imagen y a continuación, Crear imagen.
- d. Elija Create Image (Crear imagen).

11. Especifique el ID de AMI en el campo [CustomAmi](#) de la configuración del clúster y cree un clúster.

Old EC2 console

1. Busque la AWS ParallelCluster AMI que corresponda a la específica Región de AWS que utiliza. Para encontrarlo, puede usar el [pcluster list-official-images](#) comando con el `--region` parámetro para seleccionar los `--architecture` parámetros Región de AWS y `--os` y filtrar específicos para la AMI deseada con el sistema operativo y la arquitectura que desee usar. De la salida, puede recuperar el ID de imagen del EC2.
2. [Inicie sesión en la consola Amazon EC2 AWS Management Console y ábrala en https://console.aws.amazon.com/ec2/.](https://console.aws.amazon.com/ec2/)
3. En el panel de navegación, elija Imágenes y a continuación AMI. Defina el filtro para imágenes públicas y busque el ID de imagen EC2 recuperado, seleccione la AMI y elija Launch.
4. Elija el tipo de instancia y seleccione Siguiente: configurar los detalles de la instancia o Revisar y lanzar para lanzar la instancia.
5. Elija Lanzar, seleccione su par de claves y lance las instancias.
6. Inicie sesión en la instancia mediante el usuario del sistema operativo y su clave de SSH. Para obtener más información, vaya a Instancias, seleccione la nueva instancia y Conectar.
7. Personalice la instancia de forma manual para cumplir con sus requisitos.
8. Ejecute el siguiente comando para preparar su instancia para la creación de la AMI:

```
sudo /usr/local/sbin/ami_cleanup.sh
```

9. En la consola EC2, elija Instances en el panel de navegación, seleccione la nueva instancia y elija Actions, Instance State y Stop.

10. Cree una AMI nueva a partir de la instancia mediante la consola EC2 o AWS CLI [cree](#) una imagen.

Desde la consola EC2

- a. Elija Instances (Instancia[s]) en el panel de navegación.
- b. Elija la instancia que creó y modificó.
- c. Elija Acciones, Imagen y a continuación, Crear imagen.
- d. Elija Create Image (Crear imagen).

11. Especifique el ID de AMI en el campo [CustomAmi](#) de la configuración del clúster y cree un clúster.

Integración de Active Directory

En este tutorial, se crea un entorno de varios usuarios. Este entorno incluye un AWS ParallelCluster que está integrado con un AWS Managed Microsoft AD (Active Directory) `corp.example.com`. Se configura un `Admin` usuario para que administre el directorio, un `ReadOnly` usuario para que lea el directorio y un `user000` usuario para que inicie sesión en el clúster. Puede usar la ruta automática o la ruta manual para crear los recursos de red, un Active Directory (AD) y la instancia EC2 que utiliza para configurar el AD. Independientemente de la ruta, la infraestructura que cree está preconfigurada para integrarse AWS ParallelCluster mediante uno de los siguientes métodos:

- LDAPS con verificación de certificados (se recomienda como la opción más segura)
- sobre LDAPS sin verificación de certificados
- LDAP

El LDAP por sí solo no proporciona cifrado. Para garantizar la transmisión segura de información potencialmente confidencial, le recomendamos encarecidamente que utilice LDAPS (LDAP sobre TLS/SSL) para los clústeres integrados con los AD. Para obtener más información, consulte [Habilitar el uso de LDAPS del lado del servidor en la Guía de administración. AWS Managed Microsoft AD](#)
AWS Directory Service

Tras crear estos recursos, proceda a configurar y crear el clúster integrado con su Active Directory (AD). Una vez que se cree el clúster, inicie sesión como el usuario que creó. Para obtener más información sobre la configuración que se crea en este tutorial, consulte [Acceso de varios usuarios a los clústeres](#) la sección [DirectoryService](#) de configuración.

En este tutorial se explica cómo crear un entorno que admita el acceso de varios usuarios a los clústeres. En este tutorial no se explica cómo crear y utilizar un AWS Directory Service AD. Los pasos que debe seguir para configurar un AWS Managed Microsoft AD en este tutorial se proporcionan únicamente con fines de prueba. No se proporcionan para reemplazar la documentación oficial y las mejores prácticas que puede encontrar en [AWS Managed Microsoft AD Simple AD](#) en la Guía de AWS Directory Service administración.

Note

Las contraseñas de los usuarios de directorios caducan según las definiciones de propiedades de la política de contraseñas de los directorios. Para obtener más información, consulte [Supported policy permissions](#) (Permisos de políticas compatibles). Para restablecer las contraseñas de los directorios con AWS ParallelCluster, consulte [Cómo restablecer una contraseña de usuario y contraseñas caducadas](#).

Note

Las direcciones IP del controlador de dominio del directorio pueden cambiar debido a los cambios en el controlador de dominio y al mantenimiento del directorio. Si ha elegido el método de creación rápida automática para crear la infraestructura de directorios, debe alinear manualmente el balanceador de cargas frente a los controladores de directorios cuando cambien las direcciones IP del directorio. Si utilizas el método de creación rápida, las direcciones IP del directorio no se alinean automáticamente con los balanceadores de carga.

Al utilizar la interfaz de línea de comandos (CLI) o API de AWS ParallelCluster, solo paga por los recursos de AWS que se crean al crear o actualizar imágenes y clústeres de AWS ParallelCluster. Para más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

La interfaz de usuario de AWS ParallelCluster se basa en una arquitectura sin servidor y, en la mayoría de los casos, se puede utilizar en la categoría de nivel AWS gratuito. Para más información, consulte [Costes de la interfaz de usuario de AWS ParallelCluster](#).

Requisitos previos

- AWS ParallelCluster [está instalado](#).
- AWS CLI está [instalado y configurado](#).

- Tiene un [EC2 key pair](#).
- Tiene un rol de IAM con los [permisos](#) necesarios para ejecutar el CLI [pcluster](#).

A medida que avance en el tutorial *inputs highlighted in red*, sustituya (*region-id-
abcdef01234567890*) por sus propios nombres e identificadores. Reemplace *0123456789012* por el ID de su Cuenta de AWS.

Paso 1: Crear la infraestructura de AD

Elija la pestaña Automatizado para crear la infraestructura de Active Directory (AD) con una plantilla de creación AWS CloudFormation rápida.

Seleccione la pestaña Manual para crear manualmente la infraestructura de AD.

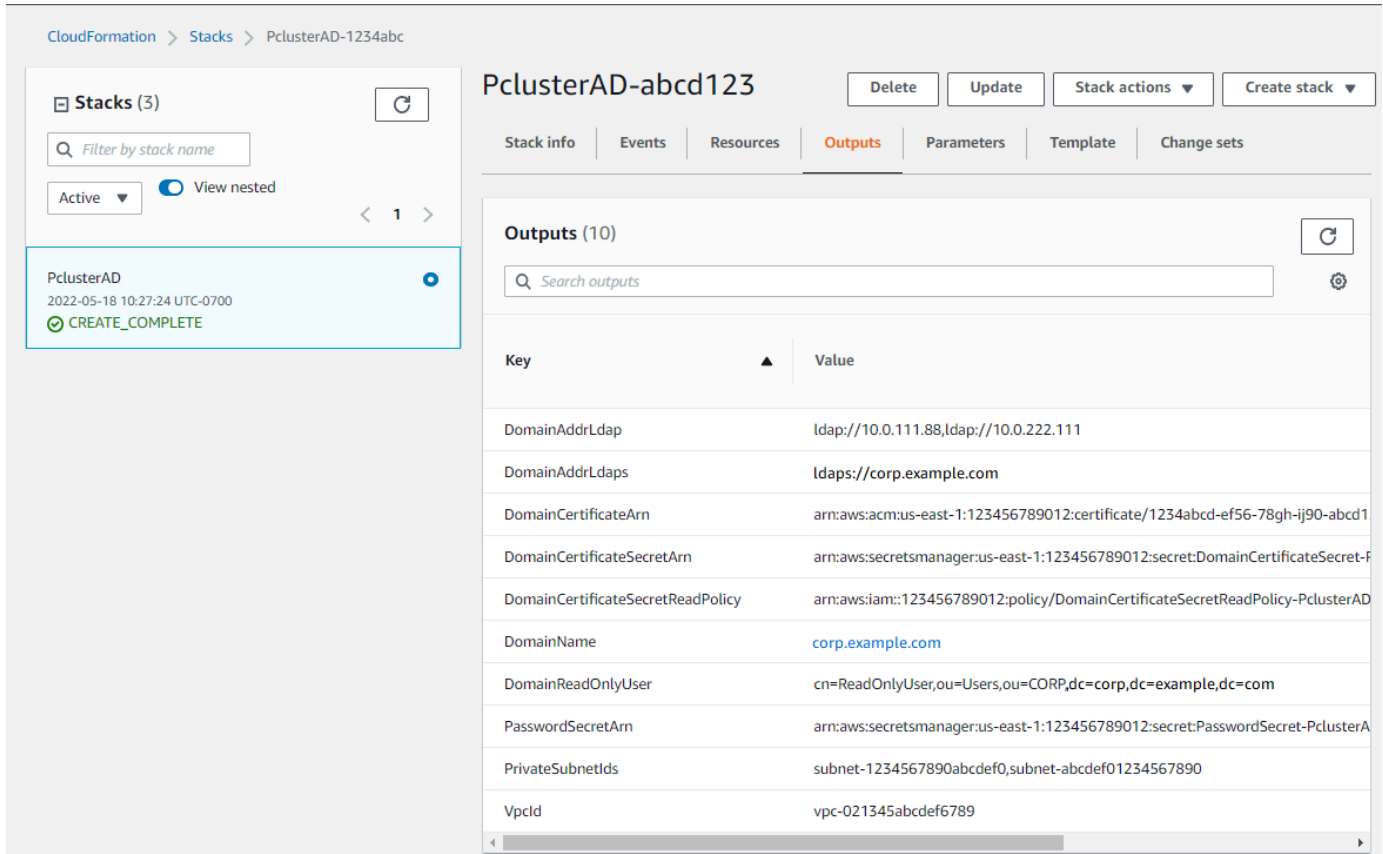
automatizar

1. Inicie sesión en la AWS Management Console.
2. Abra [CloudFormation Quick Create \(region us-east-1\)](#) para crear los siguientes recursos en la CloudFormation consola:
 - Una VPC con dos subredes y enrutamiento para acceso público, si no se especifica ninguna VPC.
 - Una AWS Managed Microsoft AD.
 - Una instancia de EC2 unida al AD que puede usar para administrar el directorio.
3. En la sección Parámetros Creación rápida de pila, proporcione valores para los parámetros siguientes:
 - AdminPassword
 - ReadOnlyPassword
 - UserPassword

Anote la contraseña. Los necesitará más adelante en este tutorial.

4. En DomainName, introduzca **corp.example.com**
5. En KeyName, escriba el nombre del par de claves EC2.
6. Elija las casillas de verificación para confirmar las capacidades de acceso en la parte inferior de la página.

7. Seleccione Crear pila.
8. Una vez que la CloudFormation pila haya alcanzado el CREATE_COMPLETE estado, seleccione la pestaña Salidas de la pila. Anote los nombres e ID de los recursos de salida porque necesitará usarlos en pasos posteriores. Los resultados proporcionan la información necesaria para crear el clúster.



The screenshot shows the AWS CloudFormation console for a stack named "PclusterAD-abcd123". The stack is in the "CREATE_COMPLETE" state. The "Outputs" tab is selected, displaying a table of 10 outputs. The table has columns for "Key" and "Value".

Key	Value
DomainAddrLdap	ldap://10.0.111.88,ldap://10.0.222.111
DomainAddrLdaps	ldaps://corp.example.com
DomainCertificateArn	arn:aws:acm:us-east-1:123456789012:certificate/1234abcd-ef56-78gh-ij90-abcd1
DomainCertificateSecretArn	arn:aws:secretsmanager:us-east-1:123456789012:secret:DomainCertificateSecret-f
DomainCertificateSecretReadPolicy	arn:aws:iam::123456789012:policy/DomainCertificateSecretReadPolicy-PclusterAD
DomainName	corp.example.com
DomainReadOnlyUser	cn=ReadOnlyUser,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
PasswordSecretArn	arn:aws:secretsmanager:us-east-1:123456789012:secret>PasswordSecret-PclusterA
PrivateSubnetIds	subnet-1234567890abcdef0,subnet-abcdef01234567890
VpcId	vpc-021345abcdef6789

9. Para completar los ejercicios [\(Opcional\) Paso 2: Administrar los usuarios y grupos de AD](#), necesitará el ID del directorio. Elija Recursos y desplácese hacia abajo para anotar el ID del directorio.
10. Continúe en [\(Opcional\) Paso 2: Administrar los usuarios y grupos de AD](#) o [Paso 1: Crear el clúster](#).

Manual

Los servidores de directorio se crean en dos subredes en dos zonas de disponibilidad diferentes dentro de una VPC.

Crea el AD

Note

- El directorio y el nombre de dominio son `corp.example.com`. El formato del nombre es CORP.
- Cambie la Admin contraseña en el script.
- La creación de Active Directory (AD) tarda al menos 15 minutos.

Utilice la siguiente secuencia de comandos de Python para crear los recursos de VPC, subredes y AD en su entorno local. Región de AWS Guarda este archivo como `ad.py` y ejecútalo.

```
import boto3
import time
from pprint import pprint

vpc_name = "PclusterVPC"
ad_domain = "corp.example.com"
admin_password = "asdfASDF1234"

ec2 = boto3.client("ec2")
ds = boto3.client("ds")
region = boto3.Session().region_name

# Create the VPC, Subnets, IGW, Routes
vpc = ec2.create_vpc(CidrBlock="10.0.0.0/16")["Vpc"]
vpc_id = vpc["VpcId"]
time.sleep(30)
ec2.create_tags(Resources=[vpc_id], Tags=[{"Key": "Name", "Value": vpc_name}])
subnet1 = ec2.create_subnet(VpcId=vpc_id, CidrBlock="10.0.0.0/17",
    AvailabilityZone=f"{region}a")["Subnet"]
subnet1_id = subnet1["SubnetId"]
time.sleep(30)
ec2.create_tags(Resources=[subnet1_id], Tags=[{"Key": "Name", "Value": f"{vpc_name}/subnet1"}])
ec2.modify_subnet_attribute(SubnetId=subnet1_id, MapPublicIpOnLaunch={"Value": True})
subnet2 = ec2.create_subnet(VpcId=vpc_id, CidrBlock="10.0.128.0/17",
    AvailabilityZone=f"{region}b")["Subnet"]
subnet2_id = subnet2["SubnetId"]
time.sleep(30)
```

```
ec2.create_tags(Resources=[subnet2_id], Tags=[{"Key": "Name", "Value": f"{vpc_name}/
subnet2"}])
ec2.modify_subnet_attribute(SubnetId=subnet2_id, MapPublicIpOnLaunch={"Value": True})
igw = ec2.create_internet_gateway()["InternetGateway"]
ec2.attach_internet_gateway(InternetGatewayId=igw["InternetGatewayId"], VpcId=vpc_id)
route_table = ec2.describe_route_tables(Filters=[{"Name": "vpc-id", "Values":
[vpc_id]}])["RouteTables"][0]
ec2.create_route(RouteTableId=route_table["RouteTableId"],
DestinationCidrBlock="0.0.0.0/0", GatewayId=igw["InternetGatewayId"])
ec2.modify_vpc_attribute(VpcId=vpc_id, EnableDnsSupport={"Value": True})
ec2.modify_vpc_attribute(VpcId=vpc_id, EnableDnsHostnames={"Value": True})

# Create the Active Directory
ad = ds.create_microsoft_ad(
    Name=ad_domain,
    Password=admin_password,
    Description="ParallelCluster AD",
    VpcSettings={"VpcId": vpc_id, "SubnetIds": [subnet1_id, subnet2_id]},
    Edition="Standard",
)
directory_id = ad["DirectoryId"]

# Wait for completion
print("Waiting for the directory to be created...")
directories = ds.describe_directories(DirectoryIds=[directory_id])
["DirectoryDescriptions"]
directory = directories[0]
while directory["Stage"] in {"Requested", "Creating"}:
    time.sleep(3)
    directories = ds.describe_directories(DirectoryIds=[directory_id])
["DirectoryDescriptions"]
    directory = directories[0]

dns_ip_addrs = directory["DnsIpAddrs"]

pprint({"directory_id": directory_id,
        "vpc_id": vpc_id,
        "subnet1_id": subnet1_id,
        "subnet2_id": subnet2_id,
        "dns_ip_addrs": dns_ip_addrs})
```

A continuación se muestra un ejemplo de salida del comando.


```
{
  "directory_id": "d-abcdef01234567890",
  "dns_ip_addrs": ["192.0.2.254", "203.0.113.237"],
  "subnet1_id": "subnet-021345abcdef6789",
  "subnet2_id": "subnet-1234567890abcdef0",
  "vpc_id": "vpc-021345abcdef6789"
}
```

Anote los nombres e ID de los recursos de salida. Los usará en pasos posteriores.

Una vez finalizada la secuencia de comandos, continúe con el siguiente paso.

Crear una instancia de EC2

New EC2 console

1. Inicie sesión en la AWS Management Console.
2. [Si no tienes un rol con las políticas enumeradas en el paso 4, abre la consola de IAM en https://console.aws.amazon.com/iam/](https://console.aws.amazon.com/iam/). De lo contrario, vaya al paso 8.
3. Cree la ResetUserPassword política y sustituya el contenido resaltado en rojo por su Región de AWS ID, ID de cuenta y ID de directorio del resultado del script que ejecutó para crear el AD.

ResetUserPassword

```
{
  "Statement": [
    {
      "Action": [
        "ds:ResetUserPassword"
      ],
      "Resource": "arn:aws:ds:region-id:123456789012:directory/d-abcdef01234567890",
      "Effect": "Allow"
    }
  ]
}
```

4. Cree un usuario de IAM con la siguiente política asociada:
 - AWSpolítica gestionada: [AmazonSSM](#). ManagedInstanceCore

- [AWS política gestionada: AmazonSSM DirectoryServiceAccess](#)
 - ResetUserPassword política
5. Abra la consola de Amazon EC2 en <https://console.aws.amazon.com/ec2/>.
 6. En el panel de Amazon EC2, elija Lanzar instancia.
 7. En Imágenes de aplicaciones y sistemas operativos, seleccione una AMI reciente de Amazon Linux 2.
 8. Para EC2 Instance type (Tipo de instancia EC2), seleccione t2.micro.
 9. En Key pair (login) (Par de claves [inicio de sesión]), elija un par de claves.
 10. En Configuración de red, elija Editar.
 11. Seleccione la VPC del directorio.
 12. Desplácese hacia abajo y seleccione Detalles avanzados.
 13. En Detalles avanzados, seleccione el directorio de unión a dominios **corp.example.com**.
 14. Para el perfil de instancia de IAM, elija el rol que creó en el paso 1 o un rol al que se adjunten las políticas enumeradas en el paso 4.
 15. En Resumen, seleccione Lanzar instancia.
 16. Anote el ID de la instancia (por ejemplo, i-1234567890abcdef0) y espere a que la instancia termine de lanzarse.
 17. Una vez que el estado de la instancia cambie a `running`, proceda con el siguiente paso.

Old EC2 console

1. Inicie sesión en la AWS Management Console.
2. [Si no tienes un rol con las políticas enumeradas en el paso 4, abre la consola de IAM en https://console.aws.amazon.com/iam/](https://console.aws.amazon.com/iam/). De lo contrario, vaya al paso 8.
3. Cree la política de ResetUserPassword. Sustituya el contenido resaltado en rojo por su Región de AWS Cuenta de AWS ID, ID y ID de directorio del resultado del script que ejecutó para crear Active Directory (AD).

ResetUserPassword

```
{
  "Statement": [
    {
      "Action": [
```

```
        "ds:ResetUserPassword"
      ],
      "Resource": "arn:aws:ds:region-id:123456789012:directory/d-abcdef01234567890",
      "Effect": "Allow"
    }
  ]
}
```

4. Cree un usuario de IAM con la siguiente política asociada:
 - AWSpolítica gestionada: [AmazonSSM ManagedInstanceCore](#)
 - AWSpolítica gestionada: [AmazonSSM DirectoryServiceAccess](#)
 - Política de ResetUserPassword
5. Abra la consola de Amazon EC2 en <https://console.aws.amazon.com/ec2/>.
6. En el panel de Amazon EC2, elija Lanzar instancia.
7. En Imágenes de aplicaciones y sistemas operativos, seleccione una AMI reciente de Amazon Linux 2.
8. En Instance type (Tipo de instancia), elija t2.micro.
9. En Key pair (login) (Par de claves [inicio de sesión]), elija un par de claves.
10. En Configuración de red, elija Editar.
11. En Configuración de red, VPC, seleccione el directorio VPC.
12. Desplázate hacia abajo y selecciona Detalles avanzados.
13. En Detalles avanzados, seleccione el directorio de unión a dominios **corp.example.com**.
14. En Detalles avanzados, perfil de instancia, elija el rol que creó en el paso 1 o un rol al que se adjunten las políticas que se enumeran en el paso 4.
15. En Resumen, elija Lanzar instancia.
16. Anote el ID de la instancia (por ejemplo i-1234567890abcdef0) y espere a que la instancia termine de lanzarse.
17. Una vez que el estado de la instancia cambie a `running`, proceda con el siguiente paso.

Une tu instancia al AD

1. Conéctese a su instancia y únase al dominio AD como **admin**.

Conéctese a la instancia y ejecute el siguiente comando.

```
$ INSTANCE_ID="i-1234567890abcdef0"
```

```
$ PUBLIC_IP=$(aws ec2 describe-instances \
--instance-ids $INSTANCE_ID \
--query "Reservations[0].Instances[0].PublicIpAddress" \
--output text)
```

```
$ ssh -i ~/.ssh/keys/keypair.pem ec2-user@$PUBLIC_IP
```

2. Instala el software necesario y únete al reino.

```
$ sudo yum -y install sssd realmd oddjob oddjob-mkhomedir adcli samba-common samba-common-tools krb5-workstation openldap-clients policycoreutils-python
```

3. Sustituya la contraseña por la nueva contraseña.

```
$ ADMIN_PW="asdfASDF1234"
```

```
$ echo $ADMIN_PW | sudo realm join -U Admin corp.example.com
Password for Admin:
```

Si lo anterior se ha realizado correctamente, se unirá al reino y continúe con el siguiente paso.

Añada usuarios a la cuenta.

1. Cree el ReadOnlyUser y un usuario adicional.

En este paso, utilizará las herramientas [adcli](#) y [openldap-clients](#) que instaló en el paso anterior.

```
$ echo $ADMIN_PW | adcli create-user -x -U Admin --domain=corp.example.com --
display-name=ReadOnlyUser ReadOnlyUser
```

```
$ echo $ADMIN_PW | adcli create-user -x -U Admin --domain=corp.example.com --
display-name=user000 user000
```

2. Compruebe que se hayan creado los usuarios:

Las direcciones IP DNS del directorio son salidas del script de Python.

```
$ DIRECTORY_IP="192.0.2.254"
```

```
$ ldapsearch -x -h $DIRECTORY_IP -D Admin -w $ADMIN_PW -b
"cn=ReadOnlyUser,ou=Users,ou=CORP,dc=corp,dc=example,dc=com"
```

```
$ ldapsearch -x -h $DIRECTORY_IP -D Admin -w $ADMIN_PW -b
"cn=user000,ou=Users,ou=CORP,dc=corp,dc=example,dc=com"
```

De forma predeterminada, cuando se crea un usuario con `adcli`, el usuario está deshabilitado.

3. Restablezca y active las contraseñas de los usuarios desde su máquina local:

No cierre todavía la sesión de su instancia EC2;

Note

- `ro-password` es la contraseña de `ReadOnlyUser`, recuperada de AWS Secrets Manager.
- `user-password` es la contraseña de un usuario del clúster que se proporciona cuando te conectas (ssh) al clúster.

`directory-id` Es una salida del script de Python.

```
$ DIRECTORY_ID="d-abcdef01234567890"
```

```
$ aws ds reset-user-password \
--directory-id $DIRECTORY_ID \
--user-name "ReadOnlyUser" \
--new-password "ro-password" \
--region "region-id"
```

```
$ aws ds reset-user-password \
--directory-id $DIRECTORY_ID \
--user-name "user000" \
```

```
--new-password "user-p@ssw0rd" \  
--region "region-id"
```

- Añade la contraseña a un secreto de Secrets Manager.

Ahora que ha creado una contraseña `ReadOnlyUser` y la ha establecido, guárdela en un secreto que se AWS ParallelCluster utilice para validar los inicios de sesión.

Usa Secrets Manager para crear un nuevo secreto que contenga la contraseña `ReadOnlyUser` como valor. El formato del valor secreto debe ser solo texto plano (no formato JSON). Toma nota del ARN secreto para futuros pasos.

```
$ aws secretsmanager create-secret --name "ADSecretPassword" \  
--region region_id \  
--secret-string "ro-p@ssw0rd" \  
--query ARN \  
--output text  
arn:aws:secretsmanager:region-id:123456789012:secret:ADSecretPassword-1234
```

Configuración de LDAPS con verificación de certificados (recomendada)

Anote los ID de recursos. Los usará en pasos posteriores.

- Genere el certificado de dominio de forma local.

```
$ PRIVATE_KEY="corp-example-com.key"  
CERTIFICATE="corp-example-com.crt"  
printf ".\n.\n.\n.\n.\n.\ncorp.example.com\n.\n" | openssl req -x509 -sha256 -nodes -  
newkey rsa:2048 -keyout $PRIVATE_KEY -days 365 -out $CERTIFICATE
```

- Guarde el certificado en Secrets Manager para poder recuperarlo del clúster más adelante.

```
$ aws secretsmanager create-secret --name example-cert \  
--secret-string file://$CERTIFICATE \  
--region region-id  
{  
  "ARN": "arn:aws:secretsmanager:region-id:123456789012:secret:example-  
cert-123abc",  
  "Name": "example-cert",  
  "VersionId": "14866070-092a-4d5a-bcdd-9219d0566b9c"  
}
```

3. Agregue la siguiente política al rol de IAM que creó para unir la instancia de EC2 con el dominio de AD.

PutDomainCertificateSecrets

```
{
  "Statement": [
    {
      "Action": [
        "secretsmanager:PutSecretValue"
      ],
      "Resource": [
        "arn:aws:secretsmanager:region-id:123456789012:secret:example-cert-123abc",
      ],
      "Effect": "Allow"
    }
  ]
}
```

4. Importe el certificado a AWS Certificate Manager (ACM).

```
$ aws acm import-certificate --certificate fileb://$CERTIFICATE \
  --private-key fileb://$PRIVATE_KEY \
  --region region-id
{
  "CertificateArn": "arn:aws:acm:region-
id:123456789012:certificate/343db133-490f-4077-b8d4-3da5bfd89e72"
}
```

5. Cree el balanceador de carga que se coloca delante de los puntos finales de Active Directory.

```
$ aws elbv2 create-load-balancer --name CorpExampleCom-NLB \
  --type network \
  --scheme internal \
  --subnets subnet-1234567890abcdef0 subnet-021345abcdef6789 \
  --region region-id
{
  "LoadBalancers": [
    {
      "LoadBalancerArn": "arn:aws:elasticloadbalancing:region-
id:123456789012:loadbalancer/net/CorpExampleCom-NLB/3afe296bf4ba80d4",
      "DNSName": "CorpExampleCom-NLB-3afe296bf4ba80d4.elb.region-id.amazonaws.com",

```

```

    "CanonicalHostedZoneId": "Z2IF0LAFXWL04F",
    "CreatedTime": "2022-05-05T12:56:55.988000+00:00",
    "LoadBalancerName": "CorpExampleCom-NLB",
    "Scheme": "internal",
    "VpcId": "vpc-021345abcdef6789",
    "State": {
      "Code": "provisioning"
    },
    "Type": "network",
    "AvailabilityZones": [
      {
        "ZoneName": "region-idb",
        "SubnetId": "subnet-021345abcdef6789",
        "LoadBalancerAddresses": []
      },
      {
        "ZoneName": "region-ida",
        "SubnetId": "subnet-1234567890abcdef0",
        "LoadBalancerAddresses": []
      }
    ],
    "IpAddressType": "ipv4"
  }
]
}

```

6. Cree el grupo objetivo que se dirige a los puntos finales de Active Directory.

```

$ aws elbv2 create-target-group --name CorpExampleCom-Targets --protocol TCP \
  --port 389 \
  --target-type ip \
  --vpc-id vpc-021345abcdef6789 \
  --region region-id
{
  "TargetGroups": [
    {
      "TargetGroupArn": "arn:aws:elasticloadbalancing:region-
id:123456789012:targetgroup/CorpExampleCom-Targets/44577c583b695e81",
      "TargetGroupName": "CorpExampleCom-Targets",
      "Protocol": "TCP",
      "Port": 389,
      "VpcId": "vpc-021345abcdef6789",
      "HealthCheckProtocol": "TCP",
      "HealthCheckPort": "traffic-port",

```



```

    "HealthCheckEnabled": true,
    "HealthCheckIntervalSeconds": 30,
    "HealthCheckTimeoutSeconds": 10,
    "HealthyThresholdCount": 3,
    "UnhealthyThresholdCount": 3,
    "TargetType": "ip",
    "IpAddressType": "ipv4"
  }
]
}

```

7. Registre los puntos finales de Active Directory (AD) en el grupo de destino.

```

$ aws elbv2 register-targets --target-group-arn
arn:aws:elasticloadbalancing:region-id:123456789012:targetgroup/CorpExampleCom-
Targets/44577c583b695e81 \
--targets Id=192.0.2.254,Port=389 Id=203.0.113.237,Port=389 \
--region region-id

```

8. Cree el listener LB con el certificado.

```

$ aws elbv2 create-listener --load-balancer-arn
arn:aws:elasticloadbalancing:region-id:123456789012:loadbalancer/net/
CorpExampleCom-NLB/3afe296bf4ba80d4 \
--protocol TLS \
--port 636 \
--default-actions
Type=forward,TargetGroupArn=arn:aws:elasticloadbalancing:region-
id:123456789012:targetgroup/CorpExampleCom-Targets/44577c583b695e81 \
--ssl-policy ELBSecurityPolicy-TLS-1-2-2017-01 \
--certificates CertificateArn=arn:aws:acm:region-
id:123456789012:certificate/343db133-490f-4077-b8d4-3da5bfd89e72 \
--region region-id
"Listeners": [
{
  "ListenerArn": "arn:aws:elasticloadbalancing:region-id:123456789012:listener/
net/CorpExampleCom-NLB/3afe296bf4ba80d4/a8f9d97318743d4b",
  "LoadBalancerArn": "arn:aws:elasticloadbalancing:region-
id:123456789012:loadbalancer/net/CorpExampleCom-NLB/3afe296bf4ba80d4",
  "Port": 636,
  "Protocol": "TLS",
  "Certificates": [
{

```

```
    "CertificateArn": "arn:aws:acm:region-
id:123456789012:certificate/343db133-490f-4077-b8d4-3da5bfd89e72"
  }
],
"SslPolicy": "ELBSecurityPolicy-TLS-1-2-2017-01",
"DefaultActions": [
  {
    "Type": "forward",
    "TargetGroupArn": "arn:aws:elasticloadbalancing:region-
id:123456789012:targetgroup/CorpExampleCom-Targets/44577c583b695e81",
    "ForwardConfig": {
      "TargetGroups": [
        {
          "TargetGroupArn": "arn:aws:elasticloadbalancing:region-
id:123456789012:targetgroup/CorpExampleCom-Targets/44577c583b695e81"
        }
      ]
    }
  }
]
}
]
```

9. Cree la zona alojada para que el dominio se pueda detectar en la VPC del clúster.

```
$ aws route53 create-hosted-zone --name corp.example.com \
--vpc VPCRegion=region-id,VPCId=vpc-021345abcdef6789 \
--caller-reference "ParallelCluster AD Tutorial"
{
  "Location": "https://route53.amazonaws.com/2013-04-01/hostedzone/
Z09020002B5MZQNXMSJUB",
  "HostedZone": {
    "Id": "/hostedzone/Z09020002B5MZQNXMSJUB",
    "Name": "corp.example.com.",
    "CallerReference": "ParallelCluster AD Tutorial",
    "Config": {
      "PrivateZone": true
    }
  },
  "ResourceRecordSetCount": 2
},
"ChangeInfo": {
  "Id": "/change/C05533343BF3IKSORW1TQ",
  "Status": "PENDING",
```

```

    "SubmittedAt": "2022-05-05T13:21:53.863000+00:00"
  },
  "VPC": {
    "VPCRegion": "region-id",
    "VPCId": "vpc-021345abcdef6789"
  }
}

```

10. Cree un archivo denominado **recordset-change.json** con el contenido siguiente. **HostedZoneId** es el ID de zona alojada canónico del balanceador de carga.

```

{
  "Changes": [
    {
      "Action": "CREATE",
      "ResourceRecordSet": {
        "Name": "corp.example.com",
        "Type": "A",
        "Region": "region-id",
        "SetIdentifier": "example-active-directory",
        "AliasTarget": {
          "HostedZoneId": "Z2IF0LAFXWL04F",
          "DNSName": "CorpExampleCom-NLB-3afe296bf4ba80d4.elb.region-
id.amazonaws.com",
          "EvaluateTargetHealth": true
        }
      }
    }
  ]
}

```

11. Envía el cambio del conjunto de registros a la zona alojada, esta vez con el ID de la zona alojada.

```

$ aws route53 change-resource-record-sets --hosted-zone-id Z09020002B5MZQNMSJUB \
--change-batch file://recordset-change.json
{
  "ChangeInfo": {
    "Id": "/change/C0137926I56R3GC7XW2Y",
    "Status": "PENDING",
    "SubmittedAt": "2022-05-05T13:40:36.553000+00:00"
  }
}

```

```
}
```

12. Cree un Dockerfile con el siguiente contenido.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "secretsmanager:GetSecretValue"
      ],
      "Resource": [
        "arn:aws:secretsmanager:region-id:123456789012:secret:example-cert-abc123"
      ],
      "Effect": "Allow"
    }
  ]
}
```

13. Cree un archivo Dockerfile denominado **policy.json** con el siguiente contenido.

```
$ aws iam create-policy --policy-name ReadCertExample \
  --policy-document file://policy.json
{
  "Policy": {
    "PolicyName": "ReadCertExample",
    "PolicyId": "ANPAUUXUVBC42VZSI4LDY",
    "Arn": "arn:aws:iam::123456789012:policy/ReadCertExample-efg456",
    "Path": "/",
    "DefaultVersionId": "v1",
    "AttachmentCount": 0,
    "PermissionsBoundaryUsageCount": 0,
    "IsAttachable": true,
    "CreateDate": "2022-05-05T13:42:18+00:00",
    "UpdateDate": "2022-05-05T13:42:18+00:00"
  }
}
```

14. Siga los pasos indicados en [\(Opcional\) Paso 2: Administrar los usuarios y grupos de AD](#) o [Paso 1: Crear el clúster](#).

(Opcional) Paso 2: Administrar los usuarios y grupos de AD

En este paso, administra los usuarios y los grupos desde una instancia EC2 de Amazon Linux 2 que está unida al dominio Active Directory (AD).

Si ha seguido la ruta automática, reinicie e inicie sesión en la instancia unida a AD que se creó como parte de la automatización.

Si has seguido la ruta manual, reinicia e inicia sesión en la instancia que creaste y uniste al AD en los pasos anteriores.

En estos pasos, se utilizan las herramientas [adcli](#) y [openldap-clients](#) que se instalaron en la instancia como parte del paso anterior.

Inicie sesión en una instancia EC2 que esté unida al dominio AD

1. En la consola EC2, seleccione la instancia EC2 sin título que se creó en los pasos anteriores. La instancia debe tener el estado Stopped.
2. Si el estado de la instancia es Detenido, elija Estado de instancia y, a continuación, Iniciar instancia.
3. Una vez superadas las comprobaciones de estado, selecciona la instancia y elige Conectar y entrar SSH en la instancia.

Administre los usuarios y grupos al iniciar sesión en una instancia EC2 de Amazon Linux 2 que se haya unido al AD

Cuando ejecute los `adcli` comandos con la `-U "Admin"` opción, se le solicitará que introduzca la Admin contraseña de AD. Incluye la Admin contraseña de AD como parte de los `ldapsearch` comandos.

1. Crear un usuario.

```
$ adcli create-user "clusteruser" --domain "corp.example.com" -U "Admin"
```

2. Para restablecer una contraseña de usuario

```
$ aws --region "region-id" ds reset-user-password --directory-id "d-
abcdef01234567890" --user-name "clusteruser" --new-password "new-p@ssw0rd"
```

3. Crear un grupo

```
$ adcli create-group "clusterteam" --domain "corp.example.com" -U "Admin"
```

4. Adición de un usuario a un grupo

```
$ adcli add-member "clusterteam" "clusteruser" --domain "corp.example.com" -U
"Admin"
```

5. Describa los usuarios y los grupos.

Describa a todos los usuarios.

```
$ ldapsearch "(&(objectClass=user))" -x -h "192.0.2.254" -b
"DC=corp,DC=example,DC=com" -D
"CN=Admin,OU=Users,OU=CORP,DC=corp,DC=example,DC=com" -w "p@ssw0rd"
```

Describa a un usuario específico.

```
$ ldapsearch "(&(objectClass=user)(cn=clusteruser))"
-x -h "192.0.2.254" -b "DC=corp,DC=example,DC=com" -D
"CN=Admin,OU=Users,OU=CORP,DC=corp,DC=example,DC=com" -w "p@ssw0rd"
```

Describa a todos los usuarios con un patrón de nombres.

```
$ ldapsearch "(&(objectClass=user)(cn=user*))" -x -h "192.0.2.254" -b
"DC=corp,DC=example,DC=com" -D
"CN=Admin,OU=Users,OU=CORP,DC=corp,DC=example,DC=com" -w "p@ssw0rd"
```

Describa a todos los usuarios que forman parte de un grupo específico.

```
$ ldapsearch "(&(objectClass=user)
(memberOf=CN=clusterteam,OU=Users,OU=CORP,DC=corp,DC=example,DC=com))"
-x -h "192.0.2.254" -b "DC=corp,DC=example,DC=com" -D
"CN=Admin,OU=Users,OU=CORP,DC=corp,DC=example,DC=com" -w "p@ssw0rd"
```

Descripción de grupos de registros

```
$ ldapsearch "objectClass=group" -x -h "192.0.2.254" -b "DC=corp,DC=example,DC=com" -D "CN=Admin,OU=Users,OU=CORP,DC=corp,DC=example,DC=com" -w "p@ssw0rd"
```

Describir un grupo específico

```
$ ldapsearch "(&(objectClass=group)(cn=clusterteam))" -x -h "192.0.2.254" -b "DC=corp,DC=example,DC=com" -D "CN=Admin,OU=Users,OU=CORP,DC=corp,DC=example,DC=com" -w "p@ssw0rd"
```

6. Eliminación de un usuario de un grupo

```
$ adcli remove-member "clusterteam" "clusteruser" --domain "corp.example.com" -U "Admin"
```

7. Elimina un usuario.

```
$ adcli delete-user "clusteruser" --domain "corp.example.com" -U "Admin"
```

8. Eliminación de un grupo

```
$ adcli delete-group "clusterteam" --domain "corp.example.com" -U "Admin"
```

Paso 1: Crear el clúster

Si no ha salido de la instancia EC2, hágalo ahora.

El entorno está configurado para crear un clúster que pueda autenticar a los usuarios en Active Directory (AD).

Cree una configuración de clúster sencilla y proporcione los ajustes pertinentes para conectarse al AD. Para obtener más información, consulte la sección [DirectoryService](#).

Elija una de las siguientes configuraciones de clúster y cópiela en un archivo con el nombre `ldaps_config.yaml`, `ldaps_nocert_config.yaml`, `oldap_config.yaml`.

Se recomienda elegir la configuración de LDAPS con verificación de certificado. Si elige esta configuración, también debe copiar el script de arranque en un archivo que lleve el nombre. `active-`

`directory.head.post.sh` Además, debe almacenarlo en un bucket de Amazon S3, tal y como se indica en el archivo de configuración.

LDAPS con configuración de verificación de certificados (recomendado)

Note

Se deben cambiar los siguientes componentes.

- `KeyName`: Uno de sus pares de claves EC2.
- `SubnetId` / `SubnetIds`: Uno de los identificadores de subred proporcionados en el resultado de la pila de creación CloudFormation rápida (tutorial automatizado) o en el script de Python (tutorial manual).
- `Region`: La región en la que creó la infraestructura de AD.
- `DomainAddr`: Esta dirección IP es una de las direcciones DNS de su servicio de AD.
- `PasswordSecretArn`: el nombre de recurso de Amazon (ARN) del secreto que contiene la contraseña del usuario `efadmin`
- `BucketName`: El nombre del depósito que contiene el script de arranque.
- `AdditionalPolicies/Policy`: El nombre de recurso de Amazon (ARN) de la política de certificación de dominio de lectura. `ReadCertExample`
- `CustomActions/OnNodeConfigured/Args`: El nombre de recurso de Amazon (ARN) secreto que contiene la política de certificación de dominios.

Para mejorar la seguridad, sugerimos utilizar la `AllowedIps` configuración `HeadNode/Ssh/` para limitar el acceso SSH al nodo principal.

```
Region: region-id
Image:
  Os: alinux2
HeadNode:
  InstanceType: t2.micro
  Networking:
    SubnetId: subnet-abcdef01234567890
  Ssh:
    KeyName: keypair
  Iam:
```



```

AdditionalIamPolicies:
  - Policy: arn:aws:iam::123456789012:policy/ReadCertExample
S3Access:
  - BucketName: my-bucket
    EnableWriteAccess: false
    KeyName: bootstrap/active-directory/active-directory.head.post.sh
CustomActions:
  OnNodeConfigured:
    Script: s3://my-bucket/bootstrap/active-directory/active-directory.head.post.sh
    Args:
      - arn:aws:secretsmanager:region-id:123456789012:secret:example-cert-123abc
      - /opt/parallelcluster/shared/directory_service/domain-certificate.crt
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue0
      ComputeResources:
        - Name: queue0-t2-micro
          InstanceType: t2.micro
          MinCount: 1
          MaxCount: 10
      Networking:
        SubnetIds:
          - subnet-abcdef01234567890
DirectoryService:
  DomainName: corp.example.com
  DomainAddr: ldaps://corp.example.com
  PasswordSecretArn: arn:aws:secretsmanager:region-id:123456789012:secret:ADSecretPassword-1234
  DomainReadOnlyUser: cn=ReadOnlyUser,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
  LdapTlsCaCert: /opt/parallelcluster/shared/directory_service/domain-certificate.crt
  LdapTlsReqCert: hard

```

Secuencia de comandos de arranque

Tras crear el archivo de arranque y antes de subirlo a su bucket de S3, ejecute `chmod +x active-directory.head.post.sh` para conceder el permiso de AWS ParallelCluster ejecución.

```

#!/bin/bash
set -e

CERTIFICATE_SECRET_ARN="$1"
CERTIFICATE_PATH="$2"

```

```

[[ -z $CERTIFICATE_SECRET_ARN ]] && echo "[ERROR] Missing CERTIFICATE_SECRET_ARN" &&
  exit 1
[[ -z $CERTIFICATE_PATH ]] && echo "[ERROR] Missing CERTIFICATE_PATH" && exit 1

source /etc/parallelcluster/cfnconfig
REGION="${cfn_region:?}"

mkdir -p $(dirname $CERTIFICATE_PATH)
aws secretsmanager get-secret-value --region $REGION --secret-id
  $CERTIFICATE_SECRET_ARN --query SecretString --output text > $CERTIFICATE_PATH

```

LDAPS sin configuración de verificación de certificados

Note

Se deben cambiar los siguientes componentes.

- **KeyName:** Uno de sus pares de claves EC2.
- **SubnetId / SubnetIds:** Uno de los identificadores de subred que aparecen en el resultado de la pila de creación CloudFormation rápida (tutorial automático) o del script de Python (tutorial manual).
- **Region:** La región en la que creó la infraestructura de AD.
- **DomainAddr:** Esta dirección IP es una de las direcciones DNS de su servicio de AD.
- **PasswordSecretArn:** el nombre de recurso de Amazon (ARN) del secreto que contiene la contraseña del usuario eadmin

Para mejorar la seguridad, sugerimos usar la `AllowedIps` configuración `HeadNode /Ssh/` para limitar el acceso SSH al nodo principal.

```

Region: region-id
Image:
  Os: alinux2
HeadNode:
  InstanceType: t2.micro
  Networking:
    SubnetId: subnet-abcdef01234567890

```

```
Ssh:
  KeyName: keypair
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue0
      ComputeResources:
        - Name: queue0-t2-micro
          InstanceType: t2.micro
          MinCount: 1
          MaxCount: 10
      Networking:
        SubnetIds:
          - subnet-abcdef01234567890
DirectoryService:
  DomainName: corp.example.com
  DomainAddr: ldaps://corp.example.com
  PasswordSecretArn: arn:aws:secretsmanager:region-
id:123456789012:secret:ADSecretPassword-1234
  DomainReadOnlyUser: cn=ReadOnlyUser,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
  LdapTlsReqCert: never
```

Configuraciones de PHP

Note

Se deben cambiar los siguientes componentes.

- `KeyName`: Uno de sus pares de claves EC2.
- `SubnetId` / `SubnetIds`: Uno de los identificadores de subred proporcionados en el resultado de la pila de creación CloudFormation rápida (tutorial automatizado) o en el script de Python (tutorial manual).
- `Region`: La región en la que creó la infraestructura de AD.
- `DomainAddr`: Esta dirección IP es una de las direcciones DNS de su servicio de AD.
- `PasswordSecretArn`: el nombre de recurso de Amazon (ARN) del secreto que contiene la contraseña del usuario eadmin

Para mejorar la seguridad, sugerimos usar la AllowedIps configuración HeadNode /Ssh/ para limitar el acceso SSH al nodo principal.

```

Region: region-id
Image:
  Os: alinux2
HeadNode:
  InstanceType: t2.micro
  Networking:
    SubnetId: subnet-abcdef01234567890
  Ssh:
    KeyName: keypair
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: queue0
      ComputeResources:
        - Name: queue0-t2-micro
          InstanceType: t2.micro
          MinCount: 1
          MaxCount: 10
      Networking:
        SubnetIds:
          - subnet-abcdef01234567890
DirectoryService:
  DomainName: dc=corp,dc=example,dc=com
  DomainAddr: ldap://192.0.2.254,ldap://203.0.113.237
  PasswordSecretArn: arn:aws:secretsmanager:region-id:123456789012:secret:ADSecretPassword-1234
  DomainReadOnlyUser: cn=ReadOnlyUser,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
  AdditionalSssdConfigs:
    ldap_auth_disable_tls_never_use_in_production: True

```

Cree el clúster con el comando siguiente.

```

$ pcluster create-cluster --cluster-name "ad-cluster" --cluster-configuration "./ldaps_config.yaml"
{
  "cluster": {
    "clusterName": "pcluster",

```

```
"cloudformationStackStatus": "CREATE_IN_PROGRESS",
"cloudformationStackArn": "arn:aws:cloudformation:region-id:123456789012:stack/ad-
cluster/1234567-abcd-0123-def0-abcdef0123456",
"region": "region-id",
"version": 3.7.0,
"clusterStatus": "CREATE_IN_PROGRESS"
}
}
```

Paso 4: conectarse al nodo de un clúster

Puede consultar el estado del clúster con el comando siguiente.

```
$ pcluster describe-cluster -n ad-cluster --region "region-id" --query "clusterStatus"
```

El resultado es el siguiente.

```
"CREATE_IN_PROGRESS" / "CREATE_COMPLETE"
```

Cuando llegue el estado "CREATE_COMPLETE", inicie sesión con el nombre de usuario y la contraseña creados.

```
$ HEAD_NODE_IP=$(pcluster describe-cluster -n "ad-cluster" --region "region-id" --query
headNode.publicIpAddress | xargs echo)
```

```
$ ssh user000@$HEAD_NODE_IP
```

Puede iniciar sesión sin la contraseña proporcionando la SSH clave que se creó para el nuevo usuario en `/home/user000@HEAD_NODE_IP/.ssh/id_rsa`.

Si el `ssh` comando se ejecutó correctamente, se conectó correctamente al clúster como usuario autenticado para usar Active Directory (AD).

Paso 5: Eliminar

1. Desde su máquina local, elimine el clúster.

```
$ pcluster delete-cluster --cluster-name "ad-cluster" --region "region-id"
{
```

```
"cluster": {
  "clusterName": "ad-cluster",
  "cloudformationStackStatus": "DELETE_IN_PROGRESS",
  "cloudformationStackArn": "arn:aws:cloudformation:region-id:123456789012:stack/ad-cluster/1234567-abcd-0123-def0-abcdef0123456",
  "region": "region-id",
  "version": "3.7.0",
  "clusterStatus": "DELETE_IN_PROGRESS"
}
```

2. Compruebe el progreso de la eliminación del clúster.

```
$ pcluster describe-cluster --cluster-name "ad-cluster" --region "region-id" --
query "clusterStatus"
"DELETE_IN_PROGRESS"
```

Una vez que el clúster se haya eliminado correctamente, continúe con el siguiente paso.

automatizar

Elimine los recursos de Active Directory

1. Navegue hasta <https://console.aws.amazon.com/cloudwatch/>.
2. En el panel de navegación, seleccione Stacks (Pilas).
3. De la lista de pilas, elige la pila AD (por ejemplo, pcluster-ad).
4. Elija Eliminar.

Manual

1. Elimine la instancia de EC2.
 - a. En <https://console.aws.amazon.com/ec2/>, elija Instances en el panel de navegación.
 - b. De la lista de instancias, elija la que creó para añadir usuarios al directorio.
 - c. Elija Instance state (Estado de instancia) y luego Terminate instance (Terminar instancia).
2. Eliminación de la zona alojada en
 - a. Cree un archivo `recordset-delete.json`, con el siguiente contenido: En este ejemplo, `HostedZoneId` es el ID canónico de la zona hospedada del balanceador de cargas.

```
{
  "Changes": [
    {
      "Action": "DELETE",
      "ResourceRecordSet": {
        "Name": "corp.example.com",
        "Type": "A",
        "Region": "region-id",
        "SetIdentifier": "pcluster-active-directory",
        "AliasTarget": {
          "HostedZoneId": "Z2IFOLAFXWL04F",
          "DNSName": "CorpExampleCom-NLB-3afe296bf4ba80d4.elb.region-id.amazonaws.com",
          "EvaluateTargetHealth": true
        }
      }
    }
  ]
}
```

- b. Envía el cambio del conjunto de registros a la zona alojada mediante el ID de la zona alojada.

```
$ aws route53 change-resource-record-sets --hosted-zone-id Z09020002B5MZQNXSJUB \
  --change-batch file://recordset-delete.json
{
  "ChangeInfo": {
    "Id": "/change/C04853642A0TH2TJ5NLNI",
    "Status": "PENDING",
    "SubmittedAt": "2022-05-05T14:25:51.046000+00:00"
  }
}
```

- c. Eliminación de la zona alojada en

```
$ aws route53 delete-hosted-zone --id Z09020002B5MZQNXSJUB
{
  "ChangeInfo": {
    "Id": "/change/C0468051QFABTVHMDEG9",
    "Status": "PENDING",
    "SubmittedAt": "2022-05-05T14:26:13.814000+00:00"
  }
}
```

```
}  
}
```

3. Eliminar el agente de escucha.

```
$ aws elbv2 delete-listener \  
  --listener-arn arn:aws:elasticloadbalancing:region-id:123456789012:listener/net/  
  CorpExampleCom-NLB/3afe296bf4ba80d4/a8f9d97318743d4b --region region-id
```

4. Eliminar un grupo de destino

```
$ aws elbv2 delete-target-group \  
  --target-group-arn arn:aws:elasticloadbalancing:region-  
  id:123456789012:targetgroup/CorpExampleCom-Targets/44577c583b695e81 --  
  region region-id
```

5. Eliminar el equilibrador de carga.

```
$ aws elbv2 delete-load-balancer \  
  --load-balancer-arn arn:aws:elasticloadbalancing:region-  
  id:123456789012:loadbalancer/net/CorpExampleCom-NLB/3afe296bf4ba80d4 --  
  region region-id
```

6. Elimine la política que usa el clúster para leer el certificado de Secrets Manager.

```
$ aws iam delete-policy --policy-arn arn:aws:iam::123456789012:policy/  
  ReadCertExample
```

7. Elimine el secreto que contiene el certificado de dominio.

```
$ aws secretsmanager delete-secret \  
  --secret-id arn:aws:secretsmanager:region-id:123456789012:secret:example-  
  cert-123abc \  
  --region region-id  
{  
  "ARN": "arn:aws:secretsmanager:region-id:123456789012:secret:example-cert-123abc",  
  "Name": "example-cert",  
  "DeletionDate": "2022-06-04T16:27:36.183000+02:00"  
}
```

8. Elimine el certificado de ACM.

```
$ aws acm delete-certificate \  

```



```
--certificate-arn arn:aws:acm:region-id:123456789012:certificate/343db133-490f-4077-b8d4-3da5bfd89e72 --region region-id
```

9. Elimine los recursos de Active Directory (AD).

a. Obtenga los siguientes identificadores de recursos de la salida del script de Python `ad.py`:

- ID DE ANUNCIO
- ID de subred de AD
- ID DE LA VPC

b. Elimine los puentes ejecutando el siguiente comando.

```
$ aws ds delete-directory --directory-id d-abcdef0123456789 --region region-id
{
  "DirectoryId": "d-abcdef0123456789"
}
```

c. Enumere los grupos de seguridad de la VPC.

```
$ aws ec2 describe-security-groups --filters '[{"Name":"vpc-id","Values":["vpc-07614ade95ebad1bc"]}]' --region region-id
```

d. Elimine el grupo de seguridad personalizado.

```
$ aws ec2 delete-security-group --group-id sg-021345abcdef6789 --region region-id
```

e. Eliminar las subredes.

```
$ aws ec2 delete-subnet --subnet-id subnet-1234567890abcdef --region region-id
```

```
$ aws ec2 delete-subnet --subnet-id subnet-021345abcdef6789 --region region-id
```

f. describe-internet-puerta de enlaces ()

```
$ aws ec2 describe-internet-gateways \
  --filters Name=attachment.vpc-id,Values=vpc-021345abcdef6789 \
  --region region-id
{
  "InternetGateways": [
```

```

    "Attachments": [
      {
        "State": "available",
        "VpcId": "vpc-021345abcdef6789"
      }
    ],
    "InternetGatewayId": "igw-1234567890abcdef",
    "OwnerId": "123456789012",
    "Tags": []
  }
]
}

```

- g. detach-internet-puerta de enlace ()

```

$ aws ec2 detach-internet-gateway \
  --internet-gateway-id igw-1234567890abcdef \
  --vpc-id vpc-021345abcdef6789 \
  --region region-id

```

- h. - Elimina la puerta de enlace de Internet.

```

$ aws ec2 delete-internet-gateway \
  --internet-gateway-id igw-1234567890abcdef \
  --region region-id

```

- i. Eliminación de la VPC

```

$ aws ec2 delete-vpc \
  --vpc-id vpc-021345abcdef6789 \
  --region region-id

```

- j. Elimine el secreto que contiene la ReadOnlyUser contraseña.

```

$ aws secretsmanager delete-secret \
  --secret-id arn:aws:secretsmanager:region-  
id:123456789012:secret:ADSecretPassword-1234 \
  --region region-id

```

Configuración del cifrado de almacenamiento compartido con una clave AWS KMS

Aprenda a configurar una clave AWS KMS administrada por el cliente para cifrar y proteger sus datos en los sistemas de almacenamiento de archivos en clúster configurados para AWS ParallelCluster.

Al utilizar la interfaz de línea de comandos (CLI) o API de AWS ParallelCluster, solo paga por los recursos de AWS que se crean al crear o actualizar imágenes y clústeres de AWS ParallelCluster. Para obtener más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

La interfaz de usuario de AWS ParallelCluster se basa en una arquitectura sin servidor y, en la mayoría de los casos, se puede utilizar en la categoría de nivel AWS gratuito. Para obtener más información, consulte [Costes de la interfaz de usuario de AWS ParallelCluster](#).

AWS ParallelCluster admite las siguientes opciones de configuración de almacenamiento compartido:

- [SharedStorage](#) / [EbsSettings](#) / [KmsKeyId](#)
- [SharedStorage](#) / [EfsSettings](#) / [KmsKeyId](#)
- [SharedStorage](#) / [FsxLustreSettings](#) / [KmsKeyId](#)

Puede utilizar estas opciones para proporcionar una clave AWS KMS gestionada por el cliente para el cifrado del sistema de almacenamiento compartido Amazon EBS, Amazon EFS y FSx for Lustre. Para usarlos, debe crear y configurar una política de IAM para lo siguiente:

- [HeadNode](#) / [Iam](#) / [AdditionalIamPolicies](#) / [Policy](#)
- [Scheduler](#) / [SlurmQueues](#) / [Iam](#) / [AdditionalIamPolicies](#) / [Policy](#)

Requisitos previos

- AWS ParallelCluster [está instalado](#).
- AWS CLI está [instalado y configurado](#).
- Tiene un [par de claves EC2](#).
- Tiene un rol de IAM con los [permisos](#) necesarios para ejecutar el CLI [pcluster](#).

Temas

- [Cree la política de .](#)
- [Configuración y creación del clúster](#)

Cree la política de .

Cree una política.

1. Vaya a la consola de IAM: <https://console.aws.amazon.com/iam/home>.
2. Elija Políticas (Políticas).
3. Elija Create Policy (Crear política).
4. Elija la pestaña JSON y pegue la siguiente política: Asegúrese de sustituir todas las apariciones de **123456789012** por su ID Cuenta de AWS y la clave Amazon Resource Name (ARN) y Región de AWS por la suya propia.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kms:DescribeKey",
        "kms:ReEncrypt*",
        "kms:CreateGrant",
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:region-id:123456789012:key/abcd1234-ef56-gh78-ij90-abcd1234efgh5678"
      ]
    }
  ]
}
```

5. para este tutorial, escriba un nombre para la política `ParallelClusterKmsPolicy` y elija Crear política.
6. Tome nota de la política del ARN. La necesita para configurar el clúster.

Configuración y creación del clúster

El siguiente es un ejemplo de configuración de clúster que incluye un sistema de archivos compartidos de Amazon Elastic Block Store con cifrado.

```
Region: eu-west-1
Image:
  Os: alinux2
HeadNode:
  InstanceType: t2.micro
  Networking:
    SubnetId: subnet-abcdef01234567890
  Ssh:
    KeyName: my-ssh-key
  Iam:
    AdditionalIamPolicies:
      - Policy: arn:aws:iam::123456789012:policy/ParallelClusterKmsPolicy
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: q1
      ComputeResources:
        - Name: t2micro
          InstanceType: t2.micro
          MinCount: 0
          MaxCount: 10
      Networking:
        SubnetIds:
          - subnet-abcdef01234567890
      Iam:
        AdditionalIamPolicies:
          - Policy: arn:aws:iam::123456789012:policy/ParallelClusterKmsPolicy
SharedStorage:
  - MountDir: /shared/ebs1
    Name: shared-ebs1
    StorageType: Ebs
    EbsSettings:
      Encrypted: True
      KmsKeyId: abcd1234-ef56-gh78-ij90-abcd1234efgh5678
```

Sustituya los elementos en rojo por sus propios valores. A continuación, cree un clúster que utilice su clave AWS KMS para cifrar los datos en Amazon EBS.

La configuración es parecida al de los sistemas de archivos de Amazon FSx para Lustre.

La configuración SharedStorage de Amazon EFS es la siguiente.

```
...
SharedStorage:
  - MountDir: /shared/efs1
    Name: shared-efs1
    StorageType: Efs
    EfsSettings:
      Encrypted: True
      KmsKeyId: abcd1234-ef56-gh78-ij90-abcd1234efgh5678
```

La configuración SharedStorage de FSx for Lustre es la siguiente.

```
...
SharedStorage:
  - MountDir: /shared/fsx1
    Name: shared-fsx1
    StorageType: FsxLustre
    FsxLustreSettings:
      StorageCapacity: 1200
      DeploymentType: PERSISTENT_1
      PerUnitStorageThroughput: 200
      KmsKeyId: abcd1234-ef56-gh78-ij90-abcd1234efgh5678
```

Ejecución de trabajos en un clúster en modo de cola múltiple

En este tutorial se explica cómo ejecutar su primer trabajo “Hello World” en AWS ParallelCluster con el modo de [cola múltiple](#).

Al utilizar la interfaz de línea de comandos (CLI) o API de AWS ParallelCluster, solo paga por los recursos de AWS que se crean al crear o actualizar imágenes y clústeres de AWS ParallelCluster. Para obtener más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

La interfaz de usuario de AWS ParallelCluster se basa en una arquitectura sin servidor y, en la mayoría de los casos, se puede utilizar en la categoría de nivel AWS gratuito. Para obtener más información, consulte [Costes de la interfaz de usuario de AWS ParallelCluster](#).

Requisitos previos

- AWS ParallelCluster [está instalado](#).

- AWS CLI está [instalado y configurado](#).
- Tiene un [par de claves EC2](#).
- Tiene un rol de IAM con los [permisos](#) necesarios para ejecutar el CLI [pcluster](#).

Configuración de su clúster

Para comprobar que AWS ParallelCluster se ha instalado correctamente, ejecute el siguiente comando:

```
$ pcluster version
```

Para obtener más información acerca de `pcluster version`, consulte [pcluster version](#).

Este comando devuelve la versión en ejecución de AWS ParallelCluster.

A continuación, ejecute `pcluster configure` para generar un archivo de configuración básico. Siga todas las instrucciones que aparecen después de este comando.

```
$ pcluster configure --config multi-queue-mode.yaml
```

Para obtener más información acerca del comando `pcluster configure`, consulte [pcluster configure](#).

Cuando realice este paso, `multi-queue-mode.yaml` aparecerá un archivo de configuración básico con el nombre. Este archivo contiene una configuración de clúster básica.

En el siguiente paso, modificará el nuevo archivo de configuración y lanzará un clúster con varias colas.

Note

Algunas instancias que se utilizan en este tutorial no son aptas para la versión gratuita.

Para este tutorial, modifique el archivo de configuración para que coincida con la siguiente configuración. Los elementos que están resaltados en rojo representan los valores del archivo de configuración. Mantén tus propios valores.

```
Region: region-id
```

```
Image:
  Os: alinux2
HeadNode:
  InstanceType: c5.xlarge
Networking:
  SubnetId: subnet-abcdef01234567890
Ssh:
  KeyName: yourkeypair
Scheduling:
  Scheduler: slurm
  SlurmQueues:
  - Name: spot
    ComputeResources:
    - Name: c5xlarge
      InstanceType: c5.xlarge
      MinCount: 1
      MaxCount: 10
    - Name: t2micro
      InstanceType: t2.micro
      MinCount: 1
      MaxCount: 10
    Networking:
      SubnetIds:
      - subnet-abcdef01234567890
  - Name: ondemand
    ComputeResources:
    - Name: c52xlarge
      InstanceType: c5.2xlarge
      MinCount: 0
      MaxCount: 10
    Networking:
      SubnetIds:
      - subnet-021345abcdef6789
```

Cree su clúster de

Cree un clúster con un nombre `multi-queue-cluster` basado en su archivo de configuración.

```
$ pcluster create-cluster --cluster-name multi-queue-cluster --cluster-configuration
multi-queue-mode.yaml
{
  "cluster": {
    "clusterName": "multi-queue-cluster",
```



```
"cloudformationStackStatus": "CREATE_IN_PROGRESS",
"cloudformationStackArn": "arn:aws:cloudformation:eu-west-1:123456789012:stack/
multi-queue-cluster/1234567-abcd-0123-def0-abcdef0123456",
"region": "eu-west-1",
"version": "3.7.0",
"clusterStatus": "CREATE_IN_PROGRESS"
}
}
```

Para obtener más información acerca del comando `pcluster create-cluster`, consulte [pcluster create-cluster](#).

Para comprobar el estado del clúster, use el comando siguiente.

```
$ pcluster list-clusters
{
  "cluster": {
    "clusterName": "multi-queue-cluster",
    "cloudformationStackStatus": "CREATE_IN_PROGRESS",
    "cloudformationStackArn": "arn:aws:cloudformation:eu-west-1:123456789012:stack/
multi-queue-cluster/1234567-abcd-0123-def0-abcdef0123456",
    "region": "eu-west-1",
    "version": "3.7.0",
    "clusterStatus": "CREATE_IN_PROGRESS"
  }
}
```

Cuando se crea el clúster, se muestra el `clusterStatus` campo `CREATE_COMPLETE`.

Inicie sesión en el nodo director

Usa tu archivo de clave SSH privado para iniciar sesión en el nodo principal.

```
$ pcluster ssh --cluster-name multi-queue-cluster -i ~/path/to/yourkeyfile.pem
```

Para obtener más información acerca de `pcluster ssh`, consulte [pcluster ssh](#).

Después de iniciar sesión, ejecute el comando `sinfo` para comprobar que sus colas de programador se instalan y configuran.

Para obtener más información acerca de `sinfo`, consulte [sinfo](#) en la Slurmdocumentación.

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
spot*      up    infinite   18   idle~ spot-dy-c5xlarge-[1-9],spot-dy-t2micro-[1-9]
spot*      up    infinite    2   idle  spot-st-c5xlarge-1,spot-st-t2micro-1
ondemand   up    infinite   10   idle~ ondemand-dy-c52xlarge-[1-10]
```

El resultado muestra que tiene uno t2.micro y un nodo de c5.xlarge cómputo en el idle estado que están disponibles en su clúster.

Todos los demás nodos están en estado de ahorro de energía, lo que se indica con el ~ sufijo en el estado de nodo, sin que ninguna instancia de EC2 los respalde. La cola predeterminada se indica con un * sufijo después del nombre de la cola. spotes la cola de trabajos predeterminada.

Ejecute el trabajo en modo de cola múltiple

A continuación, intente ejecutar un trabajo para dormir durante un tiempo. Posteriormente, el trabajo genera su propio nombre de servidor. Asegúrese de que el usuario actual pueda ejecutar este script.

```
$ tee <<EOF hellojob.sh
#!/bin/bash
sleep 30
echo "Hello World from \$(hostname)"
EOF

$ chmod +x hellojob.sh
$ ls -l hellojob.sh
-rwxrwxr-x 1 ec2-user ec2-user 57 Sep 23 21:57 hellojob.sh
```

Envíe el trabajo mediante el comando `sbatch`. Solicite dos nodos para este trabajo con la `-N 2` opción y compruebe que el trabajo se envía correctamente. Para obtener más información acerca de `sbatch`, consulte [sbatch](#) en la documentación de Slurm.

```
$ sbatch -N 2 --wrap "srun hellojob.sh"
Submitted batch job 1
```

Ahora puede ver la cola y comprobar el estado del trabajo con el comando `squeue`. Como no especificó una cola específica, se usa la cola predeterminada (spot). Para obtener más información acerca de `squeue`, consulte [squeue](#) en la documentación Slurm.

```
$ squeue
```

```

JOBID PARTITION   NAME      USER  ST      TIME  NODES NODELIST(REASON)
   1      spot     wrap ec2-user  R      0:10     2 spot-st-c5xlarge-1,spot-st-
t2micro-1

```

El resultado muestra que el trabajo se encuentra actualmente en estado de ejecución. Espere a que finalice la tarea. Este proceso tarda por 30 segundos. Luego, corre de squeue nuevo.

```

$ squeue
JOBID PARTITION   NAME      USER  ST      TIME  NODES NODELIST(REASON)

```

Ahora que todos los trabajos de la cola han terminado, busque el archivo de salida cuyo nombre aparece `slurm-1.out` en el directorio actual.

```

$ cat slurm-1.out
Hello World from spot-st-t2micro-1
Hello World from spot-st-c5xlarge-1

```

El resultado muestra que el trabajo se ha ejecutado correctamente en los nodos `spot-st-t2micro-1` y `spot-st-c5xlarge-1`.

Ahora envíe el mismo trabajo especificando las restricciones para instancias específicas con los siguientes comandos.

```

$ sbatch -N 3 -p spot -C "[c5.xlarge*1&t2.micro*2]" --wrap "srun hellojob.sh"
Submitted batch job 2

```

Ha utilizado estos parámetros para `sbatch`:

- `-N 3`— solicita tres nodos.
- `-p spot`— envía el trabajo a la cola `spot`. También puede enviar un trabajo a la `ondemand` cola especificando `-p ondemand`.
- `-C "[c5.xlarge*1&t2.micro*2]"`— especifica las restricciones de nodo específicas para este trabajo. Esto requiere que se utilicen un `c5.xlarge` nodo y dos `t2.micro` nodos para este trabajo.

Ejecute el `sinfo` comando para ver los nodos y las colas. Las colas de entrada AWS ParallelCluster se denominan particiones de entrada en Slurm.

```

$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
spot*      up    infinite   1  alloc# spot-dy-t2micro-1
spot*      up    infinite  17  idle~  spot-dy-c5xlarge-[2-10],spot-dy-t2micro-[2-9]
spot*      up    infinite   1  mix    spot-st-c5xlarge-1
spot*      up    infinite   1  alloc  spot-st-t2micro-1
ondemand   up    infinite  10  idle~  ondemand-dy-c52xlarge-[1-10]

```

Los nodos se están encendiendo. Esto se indica con el # sufijo en el estado del nodo. Ejecute el `squeue` comando para ver información sobre los trabajos del clúster.

```

$ squeue
JOBID PARTITION   NAME     USER ST       TIME  NODES NODELIST(REASON)
   2     spot     wrap    ec2-user CF        0:04     3  spot-dy-c5xlarge-1,spot-dy-
t2micro-1,spot-st-t2micro-1

```

Su trabajo está en el estado CF (CONFIGURING), esperando a que las instancias se amplíen y se unan al clúster.

Transcurridos unos tres minutos, los nodos estarán disponibles y el trabajo pasará al estado R (RUNNING).

```

$ squeue
JOBID PARTITION   NAME     USER ST       TIME  NODES NODELIST(REASON)
   2     spot     wrap    ec2-user R        0:07     3  spot-dy-t2micro-1,spot-st-
c5xlarge-1,spot-st-t2micro-1

```

El trabajo finaliza y los tres nodos están en ese `idle` estado.

```

$ squeue
JOBID PARTITION   NAME     USER ST       TIME  NODES NODELIST(REASON)
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
spot*      up    infinite  17  idle~  spot-dy-c5xlarge-[1-9],spot-dy-t2micro-[2-9]
spot*      up    infinite   3  idle   spot-dy-t2micro-1,spot-st-c5xlarge-1,spot-st-
t2micro-1
ondemand   up    infinite  10  idle~  ondemand-dy-c52xlarge-[1-10]

```

A continuación, cuando no quede ningún trabajo en la cola, compruébelo `slurm-2.out` en su directorio local.

```
$ cat slurm-2.out
Hello World from spot-st-t2micro-1
Hello World from spot-dy-t2micro-1
Hello World from spot-st-c5xlarge-1
```

Este es el estado final del clúster.

```
$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
spot*      up    infinite   17   idle~ spot-dy-c5xlarge-[1-9],spot-dy-t2micro-[2-9]
spot*      up    infinite    3   idle  spot-dy-t2micro-1,spot-st-c5xlarge-1,spot-st-
t2micro-1
ondemand   up    infinite   10   idle~ ondemand-dy-c52xlarge-[1-10]
```

Cuando realice la sesión en el clúster, podrá eliminarla al abandonar el clúster `pcluster delete-cluster`. Para obtener más información, consulte [pcluster list-clusters](#) y [pcluster delete-cluster](#).

```
$ pcluster list-clusters
{
  "clusters": [
    {
      "clusterName": "multi-queue-cluster",
      "cloudformationStackStatus": "CREATE_COMPLETE",
      "cloudformationStackArn": "arn:aws:cloudformation:eu-west-1:123456789012:stack/multi-queue-cluster/1234567-abcd-0123-def0-abcdef0123456",
      "region": "eu-west-1",
      "version": "3.1.4",
      "clusterStatus": "CREATE_COMPLETE"
    }
  ]
}
$ pcluster delete-cluster -n multi-queue-cluster
{
  "cluster": {
    "clusterName": "multi-queue-cluster",
    "cloudformationStackStatus": "DELETE_IN_PROGRESS",
    "cloudformationStackArn": "arn:aws:cloudformation:eu-west-1:123456789012:stack/multi-queue-cluster/1234567-abcd-0123-def0-abcdef0123456",
    "region": "eu-west-1",
    "version": "3.1.4",
    "clusterStatus": "DELETE_IN_PROGRESS"
  }
}
```

```
}  
}
```

Uso de la API AWS ParallelCluster

En este tutorial, creará y probará la API con [Amazon API Gateway](#) y una plantilla de AWS ParallelCluster CloudFormation. Luego, usas el cliente de ejemplo disponible en GitHub para usar la API. Para obtener más información sobre el uso de la API, consulte [API de AWS ParallelCluster](#).

Este tutorial fue extraído del taller de [HPC para clientes del sector público](#).

Al utilizar la interfaz de línea de comandos (CLI) o API de AWS ParallelCluster, solo paga por los recursos de AWS que se crean al crear o actualizar imágenes y clústeres de AWS ParallelCluster. Para obtener más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

La interfaz de usuario de AWS ParallelCluster se basa en una arquitectura sin servidor y, en la mayoría de los casos, se puede utilizar en la categoría de nivel AWS gratuito. Para obtener más información, consulte [Costes de la interfaz de usuario de AWS ParallelCluster](#).

Requisitos previos

- El AWS CLI está [instalado](#) y configurado en su entorno informático.
- AWS ParallelCluster está instalado en un entorno virtual Para obtener más información, consulte [Instalar AWS ParallelCluster en un entorno virtual](#).
- Tiene un [par de claves EC2](#).
- Tiene un rol de IAM con los [permisos](#) necesarios para ejecutar el CLI [pcluster](#).

Paso 1: Cree la API con Amazon API Gateway

Permanezca en su directorio de usuarios principal y active su entorno virtual:

1. Instale un útil procesador de línea de comandos JSON.

```
$ sudo yum groupinstall -y "Development Tools"  
sudo yum install -y jq python3-devel
```

2. Ejecute el siguiente comando para obtener la versión AWS ParallelCluster y asignarla a una variable de entorno.

```
$ PCLUSTER_VERSION=$(pcluster version | jq -r '.version')
echo "export PCLUSTER_VERSION=${PCLUSTER_VERSION}" |tee -a ~/.bashrc
```

3. Cree una variable de entorno y asígnele su ID de región.

```
$ export AWS_DEFAULT_REGION="us-east-1"
echo "export AWS_DEFAULT_REGION=${AWS_DEFAULT_REGION}" |tee -a ~/.bashrc
```

4. Ejecute los siguientes comandos para implementar la API.

```
API_STACK_NAME="pc-api-stack"
echo "export API_STACK_NAME=${API_STACK_NAME}" |tee -a ~/.bashrc
```

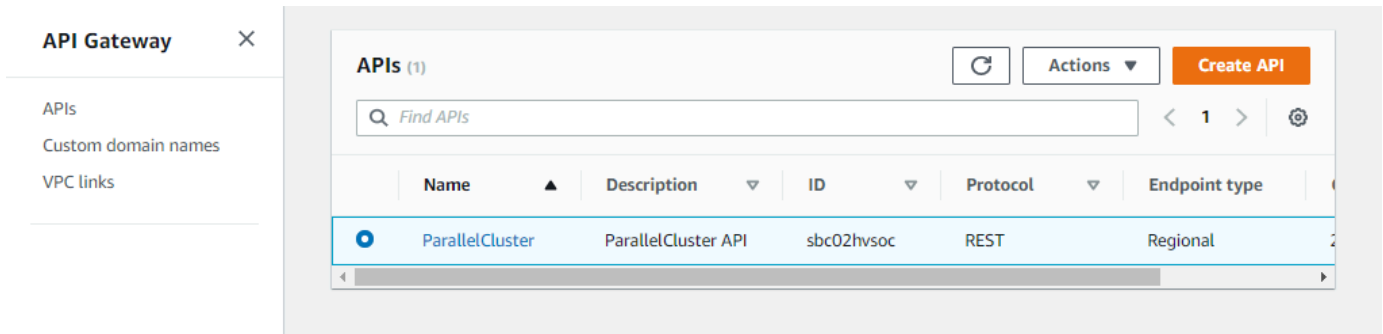
```
aws cloudformation create-stack \
  --region ${AWS_DEFAULT_REGION} \
  --stack-name ${API_STACK_NAME} \
  --template-url https://${AWS_DEFAULT_REGION}-aws-parallelcluster.s3.
  ${AWS_DEFAULT_REGION}.amazonaws.com/parallelcluster/${PCLUSTER_VERSION}/api/
  parallelcluster-api.yaml \
  --capabilities CAPABILITY_NAMED_IAM CAPABILITY_AUTO_EXPAND \
  --parameters ParameterKey=EnableIamAdminAccess,ParameterValue=true

  {
    "StackId": "arn:aws:cloudformation:us-east-1:123456789012:stack/my-api-
    stack/abcd1234-ef56-gh78-ei90-1234abcd5678"
  }
```

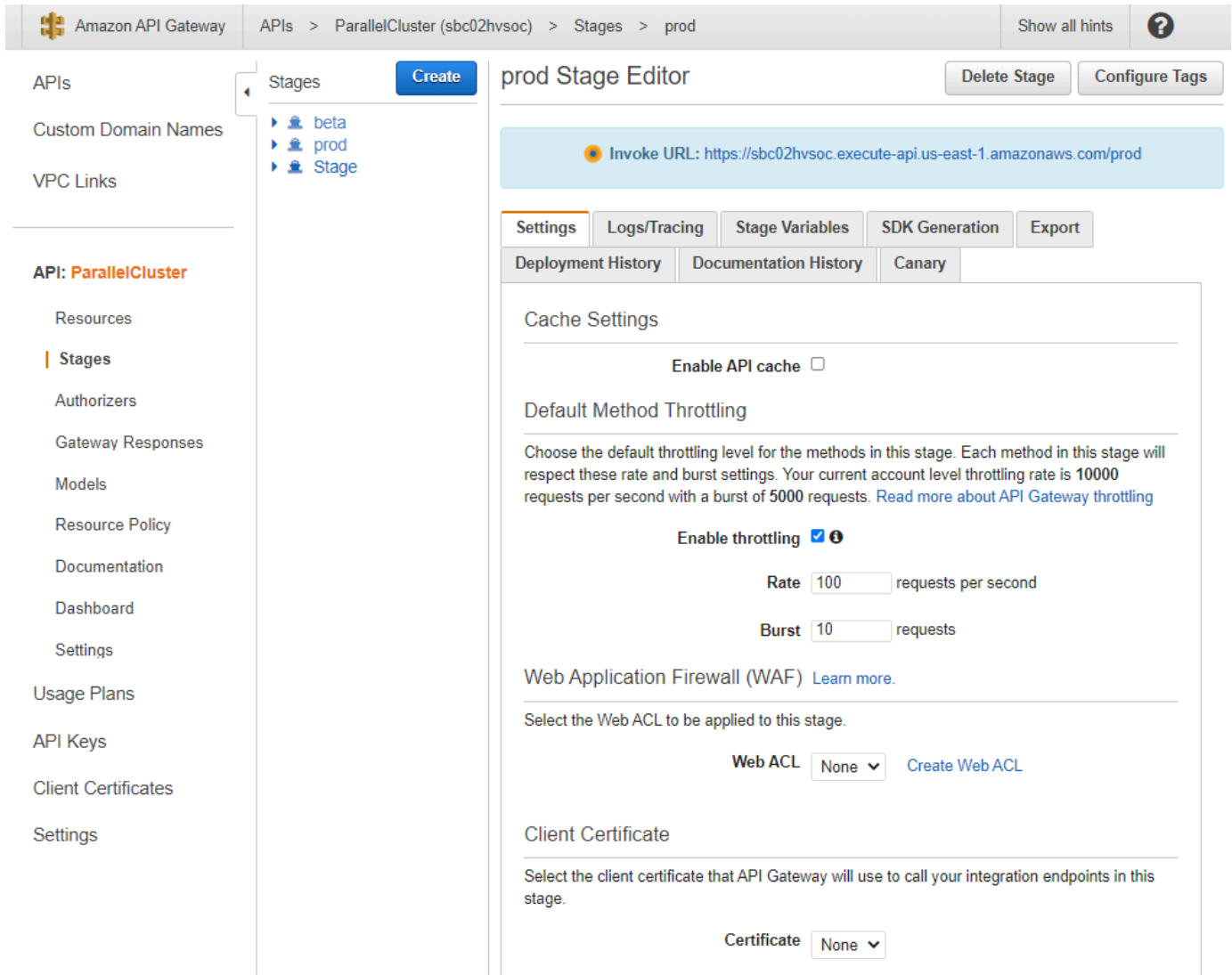
Una vez finalizada la operación, continúe con el siguiente paso.

Paso 2: Probar la API en la consola de Amazon API Gateway

1. Inicie sesión en AWS Management Console.
2. Vaya a la [consola de Amazon API Gateway](#).
3. Elija la API.



4. Elija Etapas y seleccione una etapa.



5. Anote la URL que API Gateway proporciona para acceder a su API o invocarla. Está resaltada en azul.

6. Elija Recursos y seleccione **GET** en **/clusters**.

7. Selecciona el icono de PRUEBA y, a continuación, desplázate hacia abajo y selecciona el icono de PRUEBA.

The screenshot displays the AWS API Gateway console interface. The breadcrumb navigation at the top reads: APIs > ParallelCluster (sbc02hvsoc) > Resources > /v3/clusters (ulfkw2) > GET. The main content area is titled "/v3/clusters - GET - Method Execution". On the left, a tree view shows the resource structure, with "/v3/clusters" expanded and the "GET" method selected. In the center, a vertical box labeled "Client" contains a "TEST" button with a lightning bolt icon. To the right, two panels show the execution details: "Method Request" and "Method Response".

Method Request

- Auth: AWS IAM
- ARN: arn:aws:execute-api:us-east-1:123456789012:sbc02hvsoc/*/GET/v3/clusters
- Query Strings: region, nextToken, clusterStatus

Method Response

Select an integration response.

Aparece la respuesta a su /clusters GET.

APIs > ParallelCluster (sbc02hvsoc) > Resources > /v3/clusters (ulfkw2) > GET

Show all hints ?

Resources Actions

← Method Execution /v3/clusters - GET - Method Test

Make a test call to your method. When you make a test call, API Gateway skips authorization and directly invokes your method

Path
No path parameters exist for this resource. You can define path parameters by using the syntax `{myPathParam}` in a resource path.

Request: /v3/clusters
Status: 200
Latency: 3203 ms
Response Body

Query Strings
{clusters}

Headers
{clusters}
Use a colon (:) to separate header name and value, and new lines to declare multiple headers. eg. `Accept:application/json`.

Stage Variables
No [stage variables](#) exist for this method.

Client Certificate
No client certificates have been generated.

```
{
  "clusters": [
    {
      "cloudformationStackArn": "arn:aws:cloudformation:us-east-1:123456789012:stack/test-cluster/4450d850-b684-11ec-84a7-0a047567c9f3",
      "cloudformationStackStatus": "CREATE_COMPLETE",
      "clusterName": "test-cluster",
      "clusterStatus": "CREATE_COMPLETE",
      "region": "us-east-1",
      "version": "3.1.2"
    }
  ]
}
```

Response Headers

```
{"Content-Length": "360", "X-Amzn-Trace-Id": "Root=1-62686455-c1cf243417b2721e33822ac5;Sampled=1", "Content-Type": "application/json"}
```

Logs

Paso 3: Prepare y pruebe un cliente de ejemplo para invocar la API

Clona el código AWS ParallelCluster fuente en el `api` directorio e instala las bibliotecas cliente de Python `cd`.

1.

```
$ git clone -b v${PCLUSTER_VERSION} https://github.com/aws/aws-parallelcluster aws-parallelcluster-v${PCLUSTER_VERSION}
cd aws-parallelcluster-v${PCLUSTER_VERSION}/api
```

```
$ pip3 install client/src
```

2. Vuelva al directorio de usuarios principal.
3. Exporte la URL base de API Gateway que el cliente usa cuando se ejecuta.

```
$ export PCLUSTER_API_URL=$( aws cloudformation describe-stacks
  --stack-name ${API_STACK_NAME} --query 'Stacks[0].Outputs[?
OutputKey==`ParallelClusterApiInvokeUrl`.OutputValue' --output text )
echo "export PCLUSTER_API_URL=${PCLUSTER_API_URL}" |tee -a ~/.bashrc
```

4. Exporte el nombre de un clúster que el cliente utilice para crear un clúster.

```
$ export CLUSTER_NAME="test-api-cluster"
echo "export CLUSTER_NAME=${CLUSTER_NAME}" |tee -a ~/.bashrc
```

5. Ejecute los siguientes comandos para almacenar las credenciales que el cliente de ejemplo usa para acceder a la API.

```
$ export PCLUSTER_API_USER_ROLE=$( aws cloudformation describe-
stacks --stack-name ${API_STACK_NAME} --query 'Stacks[0].Outputs[?
OutputKey==`ParallelClusterApiUserRole`.OutputValue' --output text )
echo "export PCLUSTER_API_USER_ROLE=${PCLUSTER_API_USER_ROLE}" |tee -a ~/.bashrc
```

Paso 4: Copie el script de código del cliente y ejecute las pruebas de clúster

1. Copie el siguiente código de cliente de ejemplo `test_pcluster_client.py` en su directorio de usuarios principal. El código de cliente solicita lo siguiente:
 - Cree el clúster de .
 - Describir el clúster.
 - Enumerar los clústeres.
 - Describa la flota informática.
 - Describa las instancias de un clúster.

```
# Copyright 2021 Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
#
# Permission is hereby granted, free of charge, to any person obtaining a copy of
this
```

```
# software and associated documentation files (the "Software"), to deal in the
Software
# without restriction, including without limitation the rights to use, copy,
modify,
# merge, publish, distribute, sublicense, and/or sell copies of the Software, and
to
# permit persons to whom the Software is furnished to do so.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED,
# INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
# PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
# SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
#
# Author: Evan F. Bollig (Github: bollig)

import time, datetime
import os
import pcluster_client
from pprint import pprint
from pcluster_client.api import (
    cluster_compute_fleet_api,
    cluster_instances_api,
    cluster_operations_api
)
from pcluster_client.model.create_cluster_request_content import
    CreateClusterRequestContent
from pcluster_client.model.cluster_status import ClusterStatus
region=os.environ.get("AWS_DEFAULT_REGION")

# Defining the host is optional and defaults to http://localhost
# See configuration.py for a list of all supported configuration parameters.
configuration = pcluster_client.Configuration(
    host = os.environ.get("PCLUSTER_API_URL")
)
cluster_name=os.environ.get("CLUSTER_NAME")

# Enter a context with an instance of the API client
with pcluster_client.ApiClient(configuration) as api_client:
    cluster_ops = cluster_operations_api.ClusterOperationsApi(api_client)
    fleet_ops = cluster_compute_fleet_api.ClusterComputeFleetApi(api_client)
```

```
instance_ops = cluster_instances_api.ClusterInstancesApi(api_client)

# Create cluster
build_done = False
try:
    with open('cluster-config.yaml', encoding="utf-8") as f:
        body = CreateClusterRequestContent(cluster_name=cluster_name,
cluster_configuration=f.read())
        api_response = cluster_ops.create_cluster(body, region=region)
except pcluster_client.ApiException as e:
    print("Exception when calling create_cluster: %s\n" % e)
    build_done = True
time.sleep(60)

# Confirm cluster status with describe_cluster
while not build_done:
    try:
        api_response = cluster_ops.describe_cluster(cluster_name,
region=region)
        pprint(api_response)
        if api_response.cluster_status == ClusterStatus('CREATE_IN_PROGRESS'):
            print('. . . working . . .', end='', flush=True)
            time.sleep(60)
        elif api_response.cluster_status == ClusterStatus('CREATE_COMPLETE'):
            print('READY!')
            build_done = True
        else:
            print('ERROR!!!!')
            build_done = True
    except pcluster_client.ApiException as e:
        print("Exception when calling describe_cluster: %s\n" % e)

# List clusters
try:
    api_response = cluster_ops.list_clusters(region=region)
    pprint(api_response)
except pcluster_client.ApiException as e:
    print("Exception when calling list_clusters: %s\n" % e)

# DescribeComputeFleet
try:
    api_response = fleet_ops.describe_compute_fleet(cluster_name,
region=region)
    pprint(api_response)
```

```

except pcluster_client.ApiException as e:
    print("Exception when calling compute fleet: %s\n" % e)

# DescribeClusterInstances
try:
    api_response = instance_ops.describe_cluster_instances(cluster_name,
region=region)
    pprint(api_response)
except pcluster_client.ApiException as e:
    print("Exception when calling describe_cluster_instances: %s\n" % e)

```

2. Cree una configuración de clúster:

```
$ pcluster configure --config cluster-config.yaml
```

- La biblioteca API Client detecta automáticamente los detalles de configuración de las variables de entorno (por ejemplo `AWS_ACCESS_KEY_ID`, `AWS_SECRET_ACCESS_KEY`, o `AWS_SESSION_TOKEN`) o `$HOME/.aws`. El siguiente comando cambia su función de IAM actual a la designada `ParallelClusterApiUserRole`.

```
$ eval $(aws sts assume-role --role-arn ${PCLUSTER_API_USER_ROLE} --role-session-name ApiTestSession | jq -r '.Credentials | "export AWS_ACCESS_KEY_ID=\(.AccessKeyId)\nexport AWS_SECRET_ACCESS_KEY=\(.SecretAccessKey)\nexport AWS_SESSION_TOKEN=\(.SessionToken)\n"')
```

Error al que hay que prestar atención:

Si ve un error similar al siguiente, ya ha dado por sentado que el `ParallelClusterApiUserRole` y `AWS_SESSION_TOKEN` han caducado.

```

An error occurred (AccessDenied) when calling the AssumeRole operation:
User: arn:aws:sts::XXXXXXXXXXXX:assumed-role/ParallelClusterApiUserRole-XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX/ApiTestSession
is not authorized to perform: sts:AssumeRole on resource:
arn:aws:iam::XXXXXXXXXXXX:role/ParallelClusterApiUserRole-XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX

```

Elimine el rol y, a continuación, vuelva a ejecutar el `aws sts assume-role` comando para usar el `ParallelClusterApiUserRole`.

```
$ unset AWS_SESSION_TOKEN
```

```
unset AWS_SECRET_ACCESS_KEY
unset AWS_ACCESS_KEY_ID
```

Para proporcionar sus permisos de usuario actuales para el acceso a la API, debe [ampliar la Política de recursos](#).

4. Ejecute el siguiente comando para poner a prueba el cliente de ejemplo.

```
$ python3 test_pcluster_client.py
{'cluster_configuration': 'Region: us-east-1\n'
                          'Image:\n'
                          '  Os: alinux2\n'
                          'HeadNode:\n'
                          '  InstanceType: t2.micro\n'
                          '  Networking . . . :\n'
                          '    SubnetId: subnet-1234567890abcdef0\n'
                          '  Ssh:\n'
                          '    KeyName: adpc\n'
                          'Scheduling:\n'
                          '  Scheduler: slurm\n'
                          '  SlurmQueues:\n'
                          '    - Name: queue1\n'
                          '      ComputeResources:\n'
                          '        - Name: t2micro\n'
                          '          InstanceType: t2.micro\n'
                          '          MinCount: 0\n'
                          '          MaxCount: 10\n'
                          '          Networking . . . :\n'
                          '            SubnetIds:\n'
                          '              - subnet-1234567890abcdef0\n',
  'cluster_name': 'test-api-cluster'}
{'cloud_formation_stack_status': 'CREATE_IN_PROGRESS',
 'cloudformation_stack_arn': 'arn:aws:cloudformation:us-east-1:123456789012:stack/test-api-cluster/abcd1234-ef56-gh78-ij90-1234abcd5678',
 'cluster_configuration': {'url': 'https://parallelcluster-021345abcdef6789-v1-do-not-delete...'},
 'cluster_name': 'test-api-cluster',
 'cluster_status': 'CREATE_IN_PROGRESS',
 'compute_fleet_status': 'UNKNOWN',
 'creation_time': datetime.datetime(2022, 4, 28, 16, 18, 47, 972000, tzinfo=tzlocal()),
 'last_updated_time': datetime.datetime(2022, 4, 28, 16, 18, 47, 972000, tzinfo=tzlocal()),
```

```

'region': 'us-east-1',
'tags': [{'key': 'parallelcluster:version', 'value': '3.1.3'}],
'version': '3.1.3'
.
.
.
. . . working . . . {'cloud_formation_stack_status': 'CREATE_COMPLETE',
'cloudformation_stack_arn': 'arn:aws:cloudformation:us-east-1:123456789012:stack/
test-api-cluster/abcd1234-ef56-gh78-ij90-1234abcd5678',
'cluster_configuration': {'url': 'https://parallelcluster-021345abcdef6789-v1-do-
not-delete...'},
'cluster_name': 'test-api-cluster',
'cluster_status': 'CREATE_COMPLETE',
'compute_fleet_status': 'RUNNING',
'creation_time': datetime.datetime(2022, 4, 28, 16, 18, 47, 972000,
tzinfo=tzlocal()),
'head_node': {'instance_id': 'i-abcdef01234567890',
'instance_type': 't2.micro',
'launch_time': datetime.datetime(2022, 4, 28, 16, 21, 46,
tzinfo=tzlocal()),
'private_ip_address': '172.31.27.153',
'public_ip_address': '52.90.156.51',
'state': 'running'},
'last_updated_time': datetime.datetime(2022, 4, 28, 16, 18, 47, 972000,
tzinfo=tzlocal()),
'region': 'us-east-1',
'tags': [{'key': 'parallelcluster:version', 'value': '3.1.3'}],
'version': '3.1.3'}
READY!

```

Paso 5: Copie el script de código del cliente y elimine el clúster

1. Copie el siguiente código de cliente de ejemplo en `delete_cluster_client.py`. El código de cliente solicita la eliminación del clúster.

```

# Copyright 2021 Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
#
# Permission is hereby granted, free of charge, to any person obtaining a copy of
# this
# software and associated documentation files (the "Software"), to deal in the
# Software

```



```
# without restriction, including without limitation the rights to use, copy,
# modify,
# merge, publish, distribute, sublicense, and/or sell copies of the Software, and
# to
# permit persons to whom the Software is furnished to do so.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
# IMPLIED,
# INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
# PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
# COPYRIGHT
# HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
# SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
#
# Author: Evan F. Bollig (Github: bollig)

import time, datetime
import os
import pcluster_client
from pprint import pprint
from pcluster_client.api import (
    cluster_compute_fleet_api,
    cluster_instances_api,
    cluster_operations_api
)
from pcluster_client.model.create_cluster_request_content import
    CreateClusterRequestContent
from pcluster_client.model.cluster_status import ClusterStatus
region=os.environ.get("AWS_DEFAULT_REGION")

# Defining the host is optional and defaults to http://localhost
# See configuration.py for a list of all supported configuration parameters.
configuration = pcluster_client.Configuration(
    host = os.environ.get("PCLUSTER_API_URL")
)
cluster_name=os.environ.get("CLUSTER_NAME")

# Enter a context with an instance of the API client
with pcluster_client.ApiClient(configuration) as api_client:
    cluster_ops = cluster_operations_api.ClusterOperationsApi(api_client)

    # Delete the cluster
    gone = False
```

```

try:
    api_response = cluster_ops.delete_cluster(cluster_name, region=region)
except pcluster_client.ApiException as e:
    print("Exception when calling delete_cluster: %s\n" % e)
time.sleep(60)

# Confirm cluster status with describe_cluster
while not gone:
    try:
        api_response = cluster_ops.describe_cluster(cluster_name,
region=region)
        pprint(api_response)
        if api_response.cluster_status == ClusterStatus('DELETE_IN_PROGRESS'):
            print('. . . working . . .', end='', flush=True)
            time.sleep(60)
    except pcluster_client.ApiException as e:
        gone = True
        print("DELETE COMPLETE or Exception when calling describe_cluster: %s
\n" % e)

```

2. Ejecute los siguientes comandos para eliminar el clúster de .

```

$ python3 delete_cluster_client.py
{'cloud_formation_stack_status': 'DELETE_IN_PROGRESS',
'cloudformation_stack_arn': 'arn:aws:cloudformation:us-east-1:123456789012:stack/
test-api-cluster/abcd1234-ef56-gh78-ij90-1234abcd5678',
'cluster_configuration': {'url': 'https://parallelcluster-021345abcdef6789-v1-do-
not-delete...'},
'cluster_name': 'test-api-cluster',
'cluster_status': 'DELETE_IN_PROGRESS',
'compute_fleet_status': 'UNKNOWN',
'creation_time': datetime.datetime(2022, 4, 28, 16, 50, 47, 943000,
tzinfo=tzlocal()),
'head_node': {'instance_id': 'i-abcdef01234567890',
'instance_type': 't2.micro',
'launch_time': datetime.datetime(2022, 4, 28, 16, 53, 48,
tzinfo=tzlocal()),
'private_ip_address': '172.31.17.132',
'public_ip_address': '34.201.100.37',
'state': 'running'},
'last_updated_time': datetime.datetime(2022, 4, 28, 16, 50, 47, 943000,
tzinfo=tzlocal()),
'region': 'us-east-1',
'tags': [{'key': 'parallelcluster:version', 'value': '3.1.3'}],

```

```
'version': '3.1.3'}
.
.
.
. . . working . . . {'cloud_formation_stack_status': 'DELETE_IN_PROGRESS',
'cloudformation_stack_arn': 'arn:aws:cloudformation:us-east-1:123456789012:stack/
test-api-cluster/abcd1234-ef56-gh78-ij90-1234abcd5678',
'cluster_configuration': {'url': 'https://parallelcluster-021345abcdef6789-v1-do-
not-delete...'},
'cluster_name': 'test-api-cluster',
'cluster_status': 'DELETE_IN_PROGRESS',
'compute_fleet_status': 'UNKNOWN',
'creation_time': datetime.datetime(2022, 4, 28, 16, 50, 47, 943000,
tzinfo=tzlocal()),
'last_updated_time': datetime.datetime(2022, 4, 28, 16, 50, 47, 943000,
tzinfo=tzlocal()),
'region': 'us-east-1',
'tags': [{'key': 'parallelcluster:version', 'value': '3.1.3'}],
'version': '3.1.3'}
. . . working . . . DELETE COMPLETE or Exception when calling describe_cluster:
(404)
Reason: Not Found
.
.
.
HTTP response body: {"message": "Cluster 'test-api-cluster' does not exist or
belongs to an incompatible ParallelCluster major version."}
```

3. Una vez finalizadas las pruebas, desactive las variables de entorno.

```
$ unset AWS_SESSION_TOKEN
unset AWS_SECRET_ACCESS_KEY
unset AWS_ACCESS_KEY_ID
```

Paso 6: limpiar

Puede utilizar la AWS Management Console o AWS CLI para eliminar su API.

1. En la consola AWS CloudFormation, seleccione la pila de API y a continuación, seleccione Eliminar.
2. Ejecute el siguiente comando si usa AWS CLI.

Uso de AWS CloudFormation.

```
$ aws cloudformation delete-stack --stack-name ${API_STACK_NAME}
```

Crear un clúster con contabilidad Slurm

Obtenga información sobre cómo configurar y crear un clúster de contabilidad Slurm. Para obtener más información, consulte [Slurmcontabilidad con AWS ParallelCluster](#).

Al utilizar la interfaz de línea de AWS ParallelCluster comandos (CLI) o la API, solo paga por los AWS recursos que se crean al crear o actualizar AWS ParallelCluster imágenes y clústeres. Para obtener más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

La AWS ParallelCluster interfaz de usuario se basa en una arquitectura sin servidor y, en la mayoría de los casos, se puede utilizar en la categoría de capa AWS gratuita. Para obtener más información, consulte [Costes de la interfaz de usuario de AWS ParallelCluster](#).

En este tutorial, utilizará una [plantilla de creación CloudFormation rápida \(us-east-1\) para crear una base de datos sin servidor para MySQL](#). [Amazon Aurora](#) La plantilla indica CloudFormation que se creen todos los componentes necesarios para implementar una base de datos Amazon Aurora sin servidor en la misma VPC que el clúster. La plantilla también crea una configuración básica de red y seguridad para la conexión entre el clúster y la base de datos.

Note

[A partir de la versión 3.3.0, AWS ParallelCluster admite la Slurm contabilidad con el parámetro de configuración del clúster/Base de datos. SlurmSettings](#)

Note

La plantilla de creación rápida sirve de ejemplo. Esta plantilla no cubre todos los casos de uso posibles de un servidor de base de datos de contabilidad Slurm. Es su responsabilidad crear un servidor de base de datos con la configuración y la capacidad adecuadas para sus cargas de trabajo de producción.

Requisitos previos:

- AWS ParallelCluster [está instalado](#).
- AWS CLI [está instalado y configurado](#).
- Tiene un [par de claves EC2](#).
- Tiene un rol de IAM con los [permisos](#) necesarios para ejecutar el CLI [pcluster](#).
- La región en la que implemente la plantilla de creación rápida es compatible con Amazon Aurora MySQL Serverless v2. Para obtener más información, consulte [Aurora Serverless v2 con Aurora MySQL](#).

Paso 1: Crear la VPC y las subredes para AWS ParallelCluster

Para usar la CloudFormation plantilla proporcionada para la base de datos de Slurm cuentas, debe tener lista la VPC del clúster. Puede hacerlo manualmente o como parte del procedimiento [Configure y cree un clúster con la interfaz de línea de AWS ParallelCluster comandos](#). Si ya ha utilizado AWS ParallelCluster, es posible que tenga una VPC lista para la implementación del clúster y el servidor de base de datos.

Paso 2: Crear la pila de la base de datos

Utilice la [plantilla de creación CloudFormation rápida \(us-east-1\) para crear](#) una pila de bases de datos para la contabilidad. Slurm La plantilla requiere las siguientes entradas:

- Credenciales del servidor de base de datos, específicamente el nombre de usuario y la contraseña de administrador.
- Dimensionamiento del clúster sin servidor. Amazon Aurora Esto depende de la carga prevista del clúster.
- Parámetros de red, específicamente la VPC de destino y las subredes o bloques CIDR para la creación de las subredes.

Seleccione las credenciales y el tamaño adecuados para el servidor de base de datos. Para las opciones de red, debe usar la misma VPC en la que se implementa el clúster AWS ParallelCluster . Puede crear las subredes de la base de datos y pasarlas como entrada a la plantilla. O bien, proporcione dos bloques CIDR separados para las dos subredes y deje que la CloudFormation plantilla cree las dos subredes para los bloques CIDR. Asegúrese de que los bloques CIDR no se

superpongan con las subredes existentes. Si los bloques CIDR se superponen con las subredes existentes, no se puede crear la pila.

La creación del servidor de base de datos tarda varios minutos.

Paso 3: Crear un clúster con la contabilidad Slurm habilitada

La CloudFormation plantilla proporcionada genera una CloudFormation pila con algunos resultados definidos. Desde AWS Management Console, puede ver las salidas en la pestaña Salidas de la vista de CloudFormation pila. Para habilitar la contabilidad Slurm, algunas de estas salidas deben usarse en el archivo de configuración del clúster AWS ParallelCluster :

- DatabaseHost: Se utiliza para el parámetro de configuración del clúster [SlurmSettings / Database / Uri](#).
- DatabaseAdminUser: se utiliza para el valor del parámetro de configuración del clúster de [SlurmSettings/Database/UserName](#).
- DatabaseSecretArn: Se utiliza para el parámetro de configuración del clúster [SlurmSettings / Database / PasswordSecretArn](#).
- DatabaseClientSecurityGroup: Este es el grupo de seguridad que está conectado al nodo principal del clúster que se define en el parámetro de configuración [HeadNode / Networking / SecurityGroups](#)

Actualice los parámetros Database del archivo de configuración del clúster con los valores de salida. Utilice la CLI [pcluster](#) para crear el clúster.

```
$ pcluster create-cluster -n cluster-3.x -c path/to/cluster-config.yaml
```

Una vez creado el clúster, puede empezar a utilizar comandos de contabilidad Slurm como `sacctmgr` o `sacct`.

Volver a una versión anterior del documento de Systems Manager de AWS

Aprenda a volver a una versión anterior del documento de Systems Manager de AWS. Para obtener más información sobre documentos SSM, consulte [AWS Documentos de Systems Manager](#) en la AWS Guía del usuario Systems Manager.

Al utilizar la interfaz de línea de comandos (CLI) o API de AWS ParallelCluster, solo paga por los recursos de AWS que se crean al crear o actualizar imágenes y clústeres de AWS ParallelCluster. Para obtener más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

La interfaz de usuario de AWS ParallelCluster se basa en una arquitectura sin servidor y, en la mayoría de los casos, se puede utilizar en la categoría de nivel AWS gratuito. Para obtener más información, consulte [Costes de la interfaz de usuario de AWS ParallelCluster](#).

Requisitos previos:

- Un Cuenta de AWS con permisos para administrar documentos SSM.
- AWS CLI está [instalado y configurado](#).

Volver a una versión anterior del documento SSM

1. En su terminal, ejecute el siguiente comando para obtener la lista de documentos SSM existentes que le pertenecen.

```
$ aws ssm list-documents --document-filter "key=Owner,value=Self"
```

2. Volver a una versión anterior del documento SSM En este ejemplo, volvemos a una versión anterior del documento `SessionManagerRunShell`. Puede usar el documento SSM `SessionManagerRunShell` para personalizar cada sesión de shell de SSM que inicie.
 - a. Para encontrar el parámetro `DocumentVersion` para `SessionManagerRunShell`, ejecute el siguiente comando:

```
$ aws ssm describe-document --name "SSM-SessionManagerRunShell"
{
  "Document": {
    "Hash": "...",
    "HashType": "Sha256",
    "Name": "SSM-SessionManagerRunShell",
    "Owner": "123456789012",
    "CreateDate": "2023-02-20T19:04:32.390000+00:00",
    "Status": "Active",
    "DocumentVersion": "1",
    "Parameters": [
      {
        "Name": "linuxcmd",
```

```

        "Type": "String",
        "Description": "The command to run on connection...",
        "DefaultValue": "if [ -d '/opt/parallelcluster' ]; then
source /opt/parallelcluster/cfnconfig; sudo su - $cfn_cluster_user; fi; /bin/
bash"
    }
],
"PlatformTypes": [
    "Windows",
    "Linux",
    "MacOS"
],
"DocumentType": "Session",
"SchemaVersion": "1.0",
"LatestVersion": "2",
"DefaultVersion": "1",
"DocumentFormat": "JSON",
"Tags": []
}
}

```

La última versión es 2.

- b. Vuelva a la versión anterior ejecutando el siguiente comando:

```
$ aws ssm delete-document --name "SSM-SessionManagerRunShell" --document-  
version 2
```

3. Compruebe que la versión del documento se ha revertido ejecutando de nuevo el describe-document comando:

```
$ aws ssm describe-document --name "SSM-SessionManagerRunShell"
{
  "Document": {
    "Hash": "...",
    "HashType": "Sha256",
    "Name": "SSM-SessionManagerRunShell",
    "Owner": "123456789012",
    "CreateDate": "2023-02-20T19:04:32.390000+00:00",
    "Status": "Active",
    "DocumentVersion": "1",
    "Parameters": [
      {

```



```
        "Name": "linuxcmd",
        "Type": "String",
        "Description": "The command to run on connection...",
        "DefaultValue": "if [ -d '/opt/parallelcluster' ]; then source /
opt/parallelcluster/cfnconfig; sudo su - $cfn_cluster_user; fi; /bin/bash"
    }
],
"PlatformTypes": [
    "Windows",
    "Linux",
    "MacOS"
],
"DocumentType": "Session",
"SchemaVersion": "1.0",
"LatestVersion": "1",
"DefaultVersion": "1",
"DocumentFormat": "JSON",
"Tags": []
}
}
```

La última versión es 1.

Crear un clúster con AWS CloudFormation

Obtén información sobre cómo crear un clúster con un recurso AWS ParallelCluster CloudFormation personalizado. Para obtener más información, consulte [AWS CloudFormation recurso personalizado](#).

Al usarlo AWS ParallelCluster, solo pagas por los AWS recursos que se crean al crear o actualizar AWS ParallelCluster imágenes y clústeres. Para obtener más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

Requisitos previos:

- AWS CLI [Está instalado y configurado](#).
- Un [par de claves de EC2](#).
- Un rol de IAM con los [permisos](#) necesarios para ejecutar el CLI [pcluster](#).

Creación de clústeres con una pila de creación CloudFormation rápida

En este tutorial, utilizarás una pila de creación rápida para implementar una CloudFormation plantilla que cree un clúster y los siguientes recursos: AWS

- Una CloudFormation pila raíz creada mediante una pila de creación CloudFormation rápida.
- Pilas anidadas CloudFormation que incluyen políticas predeterminadas, configuración de VPC predeterminada y un proveedor de recursos personalizado.
- Un ejemplo de pila de AWS ParallelCluster clústeres y un clúster en el que puedes iniciar sesión y ejecutar trabajos.

Cree un clúster con AWS CloudFormation

1. Inicie sesión en AWS Management Console.
2. Abre el [enlace CloudFormation de creación rápida](#) para crear los siguientes recursos en la CloudFormation consola:
 - Una CloudFormation pila anidada con una VPC con una subred pública y privada para ejecutar el nodo principal del clúster y los nodos de cómputo, respectivamente.
 - Una CloudFormation pila anidada con un recurso AWS ParallelCluster personalizado para administrar el clúster.
 - Una CloudFormation pila anidada con las políticas predeterminadas para administrar el clúster.
 - Una CloudFormation pila raíz para las pilas anidadas.
 - Un AWS ParallelCluster clúster con el Slurm planificador y un número definido de nodos de procesamiento.

CloudFormation > Stacks > Create stack

Quick create stack

Template

Template URL
https://pcluster-cfn-us-east-2.s3.amazonaws.com/parallelcluster/3.5.0/templates/custom_resource/cluster-1-click.yaml

Stack description
AWS ParallelCluster CloudFormation Cluster

Stack name

Stack name
cluster-0

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

Parameters

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

AvailabilityZone
Availability zone where instances will be launched
us-east-2a

KeyName
KeyPair to login to the head node
Select AWS::EC2::KeyPair::KeyName

Capabilities

The following resource(s) require capabilities: [AWS::CloudFormation::Stack]
This template contains Identity and Access Management (IAM) resources. Check that you want to create each of these resources and that they have the minimum required permissions. In addition, they have custom names. Check that the custom names are unique within your AWS account. [Learn more](#)

For this template, AWS CloudFormation might require an unrecognized capability: {0}. Check the capabilities of these resources. [Learn more](#)

I acknowledge that AWS CloudFormation might create IAM resources with custom names.

I acknowledge that AWS CloudFormation might require the following capability: CAPABILITY_AUTO_EXPAND

Cancel Create change set **Create stack**

3. En la sección Parámetros Creación rápida de pila, proporcione valores para los parámetros siguientes:
 - a. Para KeyName, introduzca el nombre de su key pair de EC2.

- b. Para AvailabilityZone, elija una zona de disponibilidad para los nodos del clúster, por ejemplo, `us-east-1a`.
4. Elija las casillas de verificación para confirmar las capacidades de acceso en la parte inferior de la página.
5. Seleccione Crear pila.
6. Espere a que la CloudFormation pila alcance el `CREATE_COMPLETE` estado.

Creación de clústeres con la interfaz de línea de AWS CloudFormation comandos (CLI)

En este tutorial, utilizará la interfaz de línea de AWS comandos (CLI) CloudFormation para implementar una CloudFormation plantilla que cree un clúster.

Cree los siguientes AWS recursos:

- Una CloudFormation pila raíz creada mediante una pila de CloudFormation creación rápida.
- Pilas anidadas CloudFormation que incluyen políticas predeterminadas, configuración de VPC predeterminada y un proveedor de recursos personalizado.
- Un ejemplo de pila de AWS ParallelCluster clústeres y un clúster en los que puedes iniciar sesión y ejecutar trabajos.

Sustituya *las entradas resaltadas en rojo*, como el *par de claves*, por sus propios valores.

Cree un clúster con AWS CloudFormation

1. Cree una CloudFormation plantilla denominada `cluster_template.yaml` con el siguiente contenido:

```
AWSTemplateFormatVersion: '2010-09-09'
Description: > AWS ParallelCluster CloudFormation Template

Parameters:
  KeyName:
    Description: KeyPair to login to the head node
    Type: AWS::EC2::KeyPair::KeyName
```

```

AvailabilityZone:
  Description: Availability zone where instances will be launched
  Type: AWS::EC2::AvailabilityZone::Name
  Default: us-east-2a

Mappings:
  ParallelCluster:
    Constants:
      Version: 3.7.0

Resources:
  PclusterClusterProvider:
    Type: AWS::CloudFormation::Stack
    Properties:
      TemplateURL: !Sub
        - https://${AWS::Region}-aws-parallelcluster.s3.${AWS::Region}.
          ${AWS::URLSuffix}/parallelcluster/${Version}/templates/custom_resource/cluster.yaml
        - { Version: !FindInMap [ParallelCluster, Constants, Version] }

  PclusterVpc:
    Type: AWS::CloudFormation::Stack
    Properties:
      Parameters:
        PublicCIDR: 10.0.0.0/24
        PrivateCIDR: 10.0.16.0/20
        AvailabilityZone: !Ref AvailabilityZone
      TemplateURL: !Sub
        - https://${AWS::Region}-aws-parallelcluster.s3.${AWS::Region}.
          ${AWS::URLSuffix}/parallelcluster/${Version}/templates/networking/public-private-
          ${Version}.cfm.json
        - { Version: !FindInMap [ParallelCluster, Constants, Version] }

  PclusterCluster:
    Type: Custom::PclusterCluster
    Properties:
      ServiceToken: !GetAtt [ PclusterClusterProvider , Outputs.ServiceToken ]
      ClusterName: !Sub 'c-${AWS::StackName}'
      ClusterConfiguration:
        Image:
          Os: alinux2
        HeadNode:
          InstanceType: t2.medium
        Networking:
          SubnetId: !GetAtt [ PclusterVpc , Outputs.PublicSubnetId ]

```

```

    Ssh:
      KeyName: !Ref KeyName
    Scheduling:
      Scheduler: slurm
      SlurmQueues:
        - Name: queue0
          ComputeResources:
            - Name: queue0-cr0
              InstanceType: t2.micro
          Networking:
            SubnetIds:
              - !GetAtt [ PclusterVpc , Outputs.PrivateSubnetId ]
    Outputs:
      HeadNodeIp:
        Description: The Public IP address of the HeadNode
        Value: !GetAtt [ PclusterCluster, headNode.publicIpAddress ]

```

2. Ejecute el siguiente comando AWS CLI para implementar la CloudFormation pila para la creación y administración del clúster.

```

$ aws cloudformation deploy --template-file ./cluster_template.yaml \
  --stack-name mycluster \
  --parameter-overrides KeyName=keypair \
    AvailabilityZone=us-east-2b \
  --capabilities CAPABILITY_NAMED_IAM CAPABILITY_AUTO_EXPAND

```

Ve la salida CloudFormation del clúster

Ve la salida del CloudFormation clúster para obtener detalles útiles sobre el clúster. La propiedad `ValidationMessages` agregada proporciona acceso a los mensajes de validación de las operaciones de creación y actualización del clúster.

1. Navegue hasta la [CloudFormation consola](#) y seleccione la pila que incluye su recurso AWS ParallelCluster personalizado.
2. Elija los detalles de la pila y a continuación, seleccione la pestaña Salidas.

Key	Value	Description
HeadNodeIp	1.2.3.4	The Public IP address of the HeadNode
ValidationMessages	[[{"level": "WARNING", "type": "KeyPairValidator", "message": "If you do not specify a key pair, you can't connect to the instance unless you choose an AMI that is configured to allow users another way to log in"}]]	Any warnings from cluster create or update operations.

Es posible que los mensajes de validación queden truncados. Para obtener más información acerca de la recuperación de registros, consulte [AWS ParallelCluster solución de problemas](#).

Acceso al clúster

Acceso al clúster

ssh en el nodo principal del clúster.

- Una vez completada la implementación de la CloudFormation pila, obtenga la dirección IP del nodo principal con el siguiente comando:

```
$ HEAD_NODE_IP=$(aws cloudformation describe-stacks --stack-name=mycluster --query "Stacks|[0].Outputs[?OutputKey=='HeadNodeIp']|[0].OutputValue" --output=text)
```

También puede recuperar la dirección IP del nodo principal a partir del HeadNodeIp parámetro de la pestaña Salidas de la pila de clústeres de la CloudFormation consola.

Puede encontrar la dirección IP del nodo principal aquí porque se agregó en la Outputs sección de la CloudFormation plantilla de clúster, específicamente para este clúster de ejemplo.

- Conéctese al nodo principal del clúster ejecutando el siguiente comando:

```
$ ssh -i keyname.pem ec2-user@HEAD_NODE_IP
```

Limpieza

Eliminar el clúster.

1. Ejecute el siguiente comando AWS CLI para eliminar la CloudFormation pila y el clúster.

```
$ aws cloudformation delete-stack --stack-name=mycluster
```

2. Ejecute el siguiente comando para comprobar el estado de la eliminación de la pila.

```
$ aws cloudformation describe-stacks --stack-name=mycluster
```

AWS ParallelCluster Integración de la interfaz de usuario con Identity Center

El objetivo de este tutorial es demostrar cómo integrar la AWS ParallelCluster interfaz de usuario con el Centro de identidades de IAM para lograr una solución de inicio de sesión único que unifique a los usuarios de Active Directory y pueda compartirse con clústeres. AWS ParallelCluster

Al utilizar AWS ParallelCluster, solo paga por los recursos de AWS que se crean al crear o actualizar imágenes y clústeres de AWS ParallelCluster. Para más información, consulte [Servicios de AWS utilizados por AWS ParallelCluster](#).

Requisitos previos:

- [Una AWS ParallelCluster interfaz de usuario existente que se puede instalar siguiendo las instrucciones que se indican aquí.](#)
- Un Active Directory administrado existente, preferiblemente uno [con el que también se pueda integrar AWS ParallelCluster](#).

Activar IAM Identity Center

Si ya tiene un centro de identidad conectado a su AWS Managed Microsoft AD (Active Directory), puede utilizarlo y pasar a la sección Añadir su aplicación al Centro de identidades de IAM.

Si aún no tiene un centro de identidad conectado a uno AWS Managed Microsoft AD, siga los pasos que se indican a continuación para configurarlo.

Habilitar el Centro de Identidad

1. En la consola, vaya al IAM Identity Center. (Asegúrese de estar en la región en la que tiene el suyo) AWS Managed Microsoft AD.

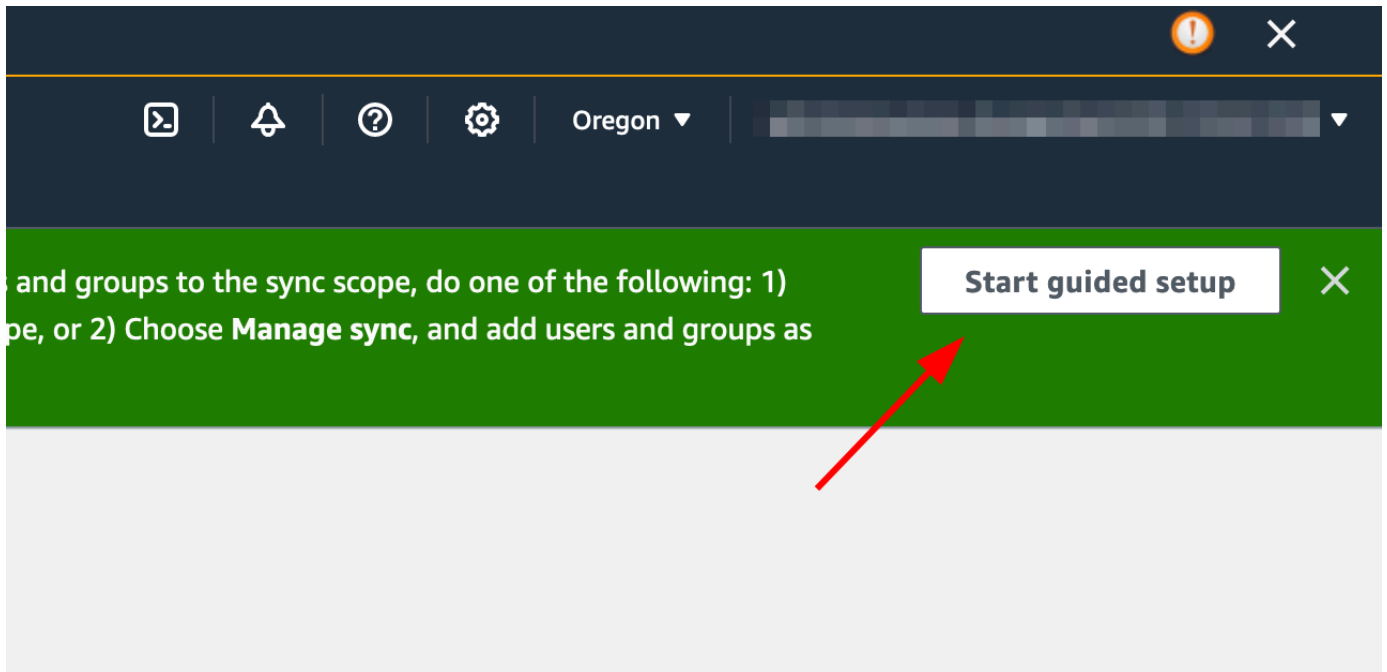
2. Haga clic en el botón Activar. Es posible que se le pregunte si desea activar las organizaciones. Se trata de un requisito para que pueda optar por activarlo. Nota: Esto enviará un correo electrónico al administrador de tu cuenta con un correo electrónico de confirmación en el que deberás seguir el enlace para confirmarlo.

Cómo conectar Identity Center a Managed AD

1. En la página siguiente, después de activar el centro de identidad, debería ver los pasos de configuración recomendados. En el paso 1, seleccione Elija su fuente de identidad.
2. En la sección Fuente de identidad, haz clic en el menú desplegable Acciones (en la parte superior derecha) y, a continuación, selecciona Cambiar fuente de identidad.
3. Seleccione Active Directory.
4. En Directorios existentes, elija su directorio.
5. Haga clic en Next (Siguiente).
6. Revisa los cambios, desplázate hasta el final, escribe ACCEPT en el cuadro de texto para confirmarlos y, a continuación, haz clic en Cambiar fuente de identidad.
7. Espera a que se completen los cambios y verás un cartel verde en la parte superior.

Sincronizar usuarios y grupos con Identity Center

1. En el banner verde, haga clic en Iniciar la configuración guiada (botón situado en la parte superior derecha)



2. En la sección Configurar asignaciones de atributos, haga clic en Siguiente
3. En la sección Configurar el alcance de la sincronización, escriba el nombre de los usuarios que desea sincronizar con el centro de identidades y, a continuación, haga clic en Agregar
4. Cuando termine de agregar usuarios y grupos, haga clic en Siguiente

Users
Groups

User

corp.pcluster.com
▼

🔍

Add

Added users and groups (4) Remove

<input type="checkbox"/>	Username / Group name	Type	Domain
<input type="checkbox"/>	user1	User	corp.pcluster.com
<input type="checkbox"/>	user2	User	corp.pcluster.com
<input type="checkbox"/>	admin1	User	corp.pcluster.com
<input type="checkbox"/>	admin2	User	corp.pcluster.com

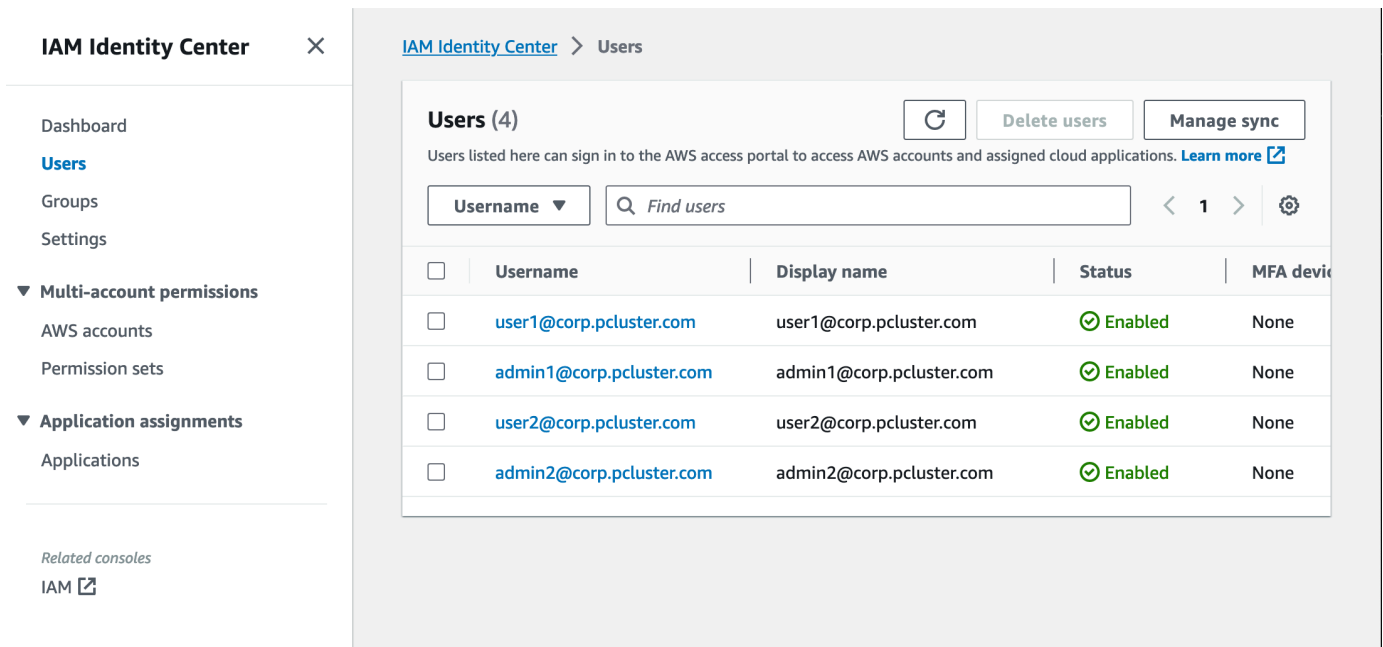
Cancel

Previous

Next

5. Revisa los cambios y, a continuación, haz clic en Guardar configuración
6. Si ves una advertencia en la siguiente pantalla sobre los usuarios que no se están sincronizando, selecciona el botón Reanudar la sincronización en la esquina superior derecha.
7. A continuación, para activar los usuarios, en la pestaña Usuarios de la izquierda, selecciona un usuario y, a continuación, haz clic en Habilitar el acceso de los usuarios > Habilitar el acceso de los usuarios

Nota: Puede que tengas que seleccionar Reanudar la sincronización si tienes un cartel de advertencia en la parte superior y esperar a que los usuarios se sincronicen (prueba con el botón de actualización para comprobar si ya están sincronizados).



The screenshot shows the IAM Identity Center console interface. On the left is a navigation menu with options like Dashboard, Users, Groups, Settings, Multi-account permissions, and Application assignments. The main content area is titled 'IAM Identity Center > Users' and displays 'Users (4)'. It includes buttons for 'Delete users' and 'Manage sync', a search bar, and a table of users.

<input type="checkbox"/>	Username	Display name	Status	MFA device
<input type="checkbox"/>	user1@corp.pcluster.com	user1@corp.pcluster.com	Enabled	None
<input type="checkbox"/>	admin1@corp.pcluster.com	admin1@corp.pcluster.com	Enabled	None
<input type="checkbox"/>	user2@corp.pcluster.com	user2@corp.pcluster.com	Enabled	None
<input type="checkbox"/>	admin2@corp.pcluster.com	admin2@corp.pcluster.com	Enabled	None

Añadir su aplicación al centro de identidad de IAM

Una vez que haya sincronizado sus usuarios con el Centro de identidades de IAM, tendrá que añadir una nueva aplicación. Esto configura qué aplicaciones habilitadas para el SSO estarán disponibles en su portal del Centro de Identidad de IAM. En este caso, añadiremos la AWS ParallelCluster interfaz de usuario como aplicación, mientras que el IAM Identity Center será el proveedor de identidad.

El siguiente paso será añadir la AWS ParallelCluster interfaz de usuario como aplicación en el IAM Identity Center. AWS ParallelCluster La interfaz de usuario es un portal web que ayuda al usuario a gestionar sus clústeres. Para obtener más información, consulte la [AWS ParallelCluster interfaz de usuario](#).

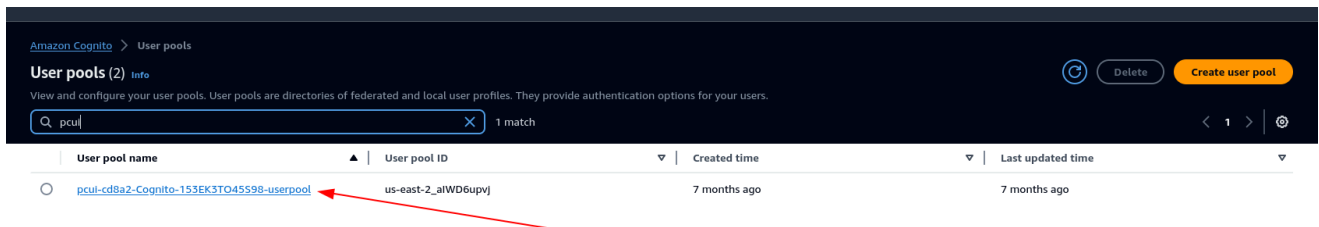
Configuración de la aplicación en Identity Center

1. En IAM Identity Center > Aplicaciones (en la barra de menú de la izquierda, haga clic en Aplicaciones)
2. Haga clic en Añadir aplicación
3. Seleccione Añadir aplicación SAML 2.0 personalizada
4. Haga clic en Siguiente
5. Seleccione el nombre para mostrar y la descripción que desee utilizar (p. ej., PCUI e AWS ParallelCluster UI)

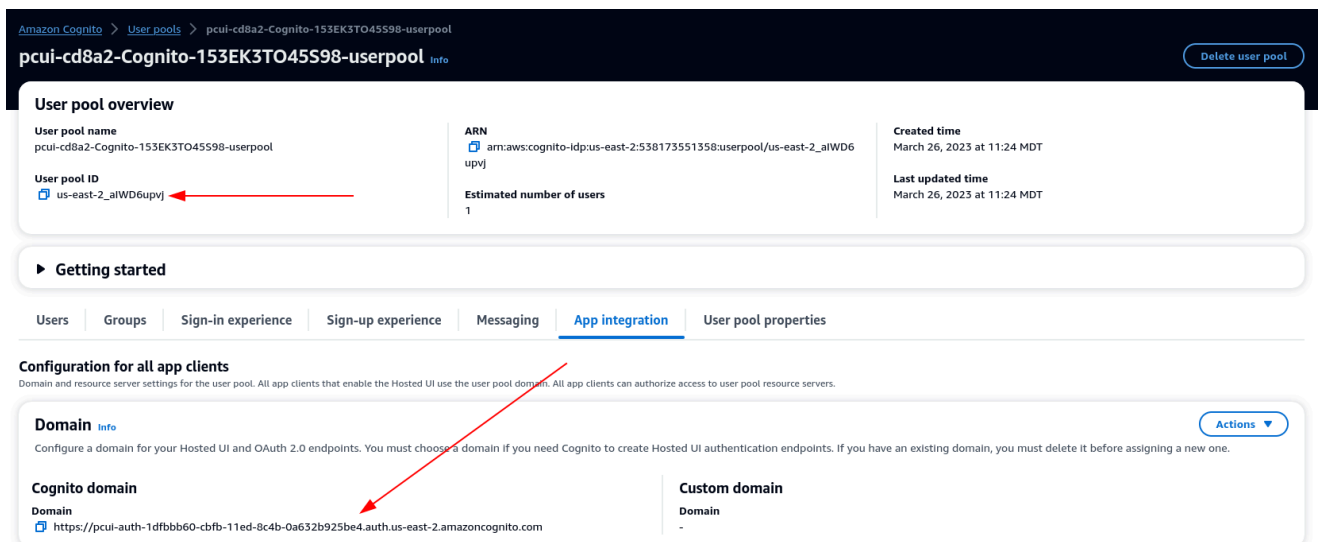
6. En la sección Metadatos del Centro de Identidad de IAM, copia el enlace al archivo de metadatos SAML del Centro de Identidad de IAM y guárdalo para más adelante, ya que se utilizará al configurar el SSO en la aplicación web
7. En Propiedades de la aplicación, en la URL de inicio de la aplicación, introduzca su dirección de PCUI. Para encontrarla, ve a la CloudFormation consola, selecciona la pila que corresponde a la PCUI (por ejemplo, parallelcluster-ui) y ve a la pestaña Salidas para buscar UIURL ParallelCluster

p. ej. <https://m2iwazsi1j.execute-api.us-east-1.amazonaws.com>

8. En Metadatos de la aplicación, elija Escribir manualmente los valores de los metadatos. A continuación, proporcione los siguientes valores.
 - a. Importante: asegúrese de reemplazar los valores del prefijo de dominio, la región y el identificador del grupo de usuarios por información específica de su entorno.
 - b. El prefijo de dominio, la región y el ID del grupo de usuarios se pueden obtener abriendo la consola Amazon Cognito > User Pools



- c. Seleccione el grupo de usuarios que corresponda a la PCUI (que tendrá un nombre de grupo de usuarios como PCUI-CD8a2-cognito-153ek3to45s98-userpool)
- d. Navegue hasta Integración de aplicaciones



9. <domain-prefix>URL del Application Assertion Consumer Service (ACS): <https://.auth.<region>.amazoncognito.com/saml2/idpresponse>

Público SAML de la aplicación: urn:amazon:cognito:sp: <userpool-id>

10. Elija Enviar. A continuación, vaya a la página de detalles de la aplicación que ha agregado.
11. Seleccione la lista desplegable Acciones y elija Editar asignaciones de atributos. A continuación, proporcione los siguientes atributos.
- Atributo de usuario en la aplicación: asunto (Nota: el asunto viene relleno previamente).
→ Se asigna a este valor de cadena o atributo de usuario del Centro de Identidad de IAM: `${user:email}`, formato: emailAddress
 - Atributo de usuario de la aplicación: correo electrónico → Se asigna a este valor de cadena o atributo de usuario del Centro de Identidad de IAM: `${user:email}`, Formato: sin especificar

Attribute mappings for PCUI

Attributes you map here become part of the SAML assertion that is sent to the application. You can choose which user attributes in your application map to corresponding user attributes in your connected directory. [Learn more](#)

User attribute in the application	Maps to this string value or user attribute in IAM Identity Center	Format	
Subject	<code>\${user:email}</code>	emailAddress	
email	<code>\${user:email}</code>	unspecified	Remove
Add new attribute mapping			

Cancel Save changes

12. Guarde los cambios.
13. Pulse el botón Asignar usuarios y, a continuación, asigne su usuario a la aplicación. Estos son los usuarios de su Active Directory que tendrán acceso a la interfaz de la PCUI.

IAM Identity Center > Applications > PCUI

PCUI

Details Actions

Icon	Display name	Description
	PCUI	AWS ParallelCluster UI

Assigned users (1) Remove access Assign Users

The following users and groups from your connected directory can access this application. [Learn more](#)

Search for an assigned user or group

User/Group name	Type
<input type="checkbox"/> admin1@corp.pcluster.com	User

Configure el centro de identidad de IAM como un IdP de SAML en su grupo de usuarios

1. En la configuración de su grupo de usuarios, seleccione Experiencia de inicio de sesión > Añadir proveedor de identidad

The screenshot shows the AWS IAM console interface for a user pool. The breadcrumb navigation is: Amazon Cognito > User pools > parallelcluster-ui-Cognito-ZQRNCGGD3MHU-userpool. The user pool name is 'parallelcluster-ui-Cognito-ZQRNCGGD3MHU-userpool' and the ID is 'us-east-1_Bgyu7LLz6'. The ARN is 'arn:aws:cognito-idp:us-east-1:538173551358:userpool/us-east-1_Bgyu7LLz6'. The estimated number of users is 1. The user pool was created on November 5, 2023 at 12:32 MST and last updated on the same date and time. The 'Getting started' section has tabs for Users, Groups, Sign-in experience (selected), Sign-up experience, Messaging, App integration, and User pool properties. The 'Sign-in experience' section includes 'Cognito user pool sign-in' and 'Federated identity provider sign-in (0)'. The 'Federated identity provider sign-in' section has a search bar and a table with columns: Identity provider, Identity provider type, Created time, and Last updated time. The table is currently empty, showing 'No identity providers' and an 'Add identity provider' button. A red arrow points to the 'Add identity provider' button in the 'Federated identity provider sign-in' section.

2. Elija un IdP de SAML
3. Para el nombre del proveedor, indique IdentityCenter
4. En Origen del documento de metadatos, seleccione Introducir la URL del punto final del documento de metadatos e indique la URL copiada durante la configuración de la aplicación en Identity Center
5. En los Atributos, para correo electrónico, elija correo electrónico

SAML

Configure a SAML 2.0 identity provider for your user pool.

Register your app with your SAML provider

To connect a SAML provider to Cognito, add your user pool as a relying party or application with your SAML 2.0 identity provider, and upload a metadata document to Cognito.

Set up SAML federation with this user pool

Provider name [Info](#)
Enter a friendly name for your SAML 2.0 identity provider.

Identifiers - optional [Info](#)
Enter identifiers for this provider. Identifiers can be used to redirect users to the correct IdP in multitenant apps.

Separate each identifier by a comma

Sign-out flow [Info](#)
 Add sign-out flow
Enable simultaneous sign-out from the SAML provider and Cognito.

Metadata document source [Info](#)
Provide a SAML metadata document. This document is issued by your SAML provider. It includes the issuer's name, expiration information, and keys that can be used to validate the response from the identity provider.

Upload metadata document
 Enter metadata document endpoint URL

Enter metadata document endpoint URL [Info](#)

Map attributes between your SAML provider and your user pool

Your required attributes are mapped to the equivalent SAML attributes. Each attribute you add must be mapped to a SAML attribute.

User pool attribute	SAML attribute
email	email

[Add another attribute](#)

[Cancel](#) [Add identity provider](#)

6. Seleccione Agregar proveedor de identidades.

Integre el IdP con el cliente de la aplicación del grupo de usuarios

1. A continuación, en la sección Integración de aplicaciones de su grupo de usuarios, elija el cliente que aparece en la lista de clientes de aplicaciones

Amazon Cognito > User pools > parallelcluster-ui-Cognito-ZQRNCGGD3MHU-userpool

parallelcluster-ui-Cognito-ZQRNCGGD3MHU-userpool Info

[Delete user pool](#)

User pool overview

User pool name parallelcluster-ui-Cognito-ZQRNCGGD3MHU-userpool	ARN arn:aws:cognito-idp:us-east-1:123456789012:userpool/us-east-1_Bgyu7Lz6	Created time November 5, 2023 at 12:32 MST
User pool ID us-east-1_Bgyu7Lz6	Estimated number of users 1	Last updated time November 5, 2023 at 12:32 MST

Getting started

Users | Groups | Sign-in experience | Sign-up experience | Messaging | **App integration** | User pool properties

Configuration for all app clients

Domain and resource server settings for the user pool. All app clients that enable the Hosted UI use the user pool domain. All app clients can authorize access to user pool resource servers.

Domain Info

Configure a domain for your Hosted UI and OAuth 2.0 endpoints. You must choose a domain if you need Cognito to create Hosted UI authentication endpoints. If you have an existing domain, you must delete it before assigning a new one.

[Actions](#)

Cognito domain	Custom domain
Domain https://pcul-auth-06f29200-7c12-11ee-b755-0e11297ecb0d.auth.us-east-1.amazoncognito.com	Domain -

Resource servers (0) Info

Configure resource servers. A resource server is a remote server that authorizes access based on OAuth 2.0 scopes in an access token.

[Edit](#) [Delete](#) [Create resource server](#)

Resource server name	Resource server identifier	Custom scopes
No resource servers		

[Create resource server](#)

App client defaults

Hosted UI customization and advanced security settings for the user pool. You can customize the Hosted UI and advanced security in app clients to override the defaults.

Hosted UI customization Info

Customize the hosted sign-up and sign-in pages to match your app's style and branding by uploading your own logo and customized CSS.

[Edit](#)

Logo -	Custom CSS -
------------------	------------------------

Advanced security Info

Configure advanced security features, including Cognito's automatic responses to suspicious user activity. Advanced security adds cost to your bill. [See pricing](#)

[Enable](#)

Status
 Disabled

App client list

The app clients that integrate your apps with your user pool. Configure client overrides to user pool default configurations, and configure Amazon Pinpoint analytics.

App clients and analytics (1) Info

Configure an app client. App clients are the user pool authentication resources attached to your app. Select an app client to configure the permitted authentication actions for an app.

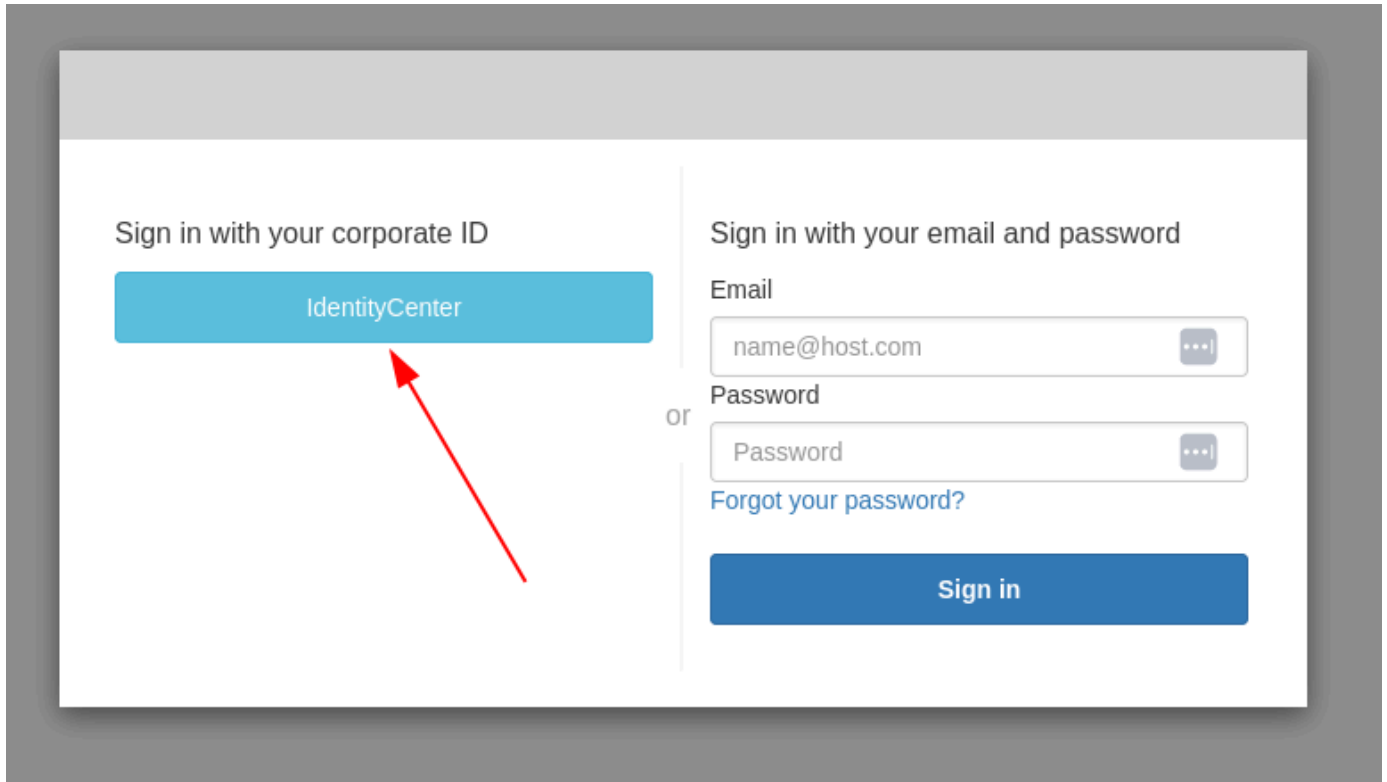
[Refresh](#) [Delete](#) [Create app client](#)

App client name	Client ID
<input type="radio"/> CognitoAppClient-ETqXbqI5wRVs	...

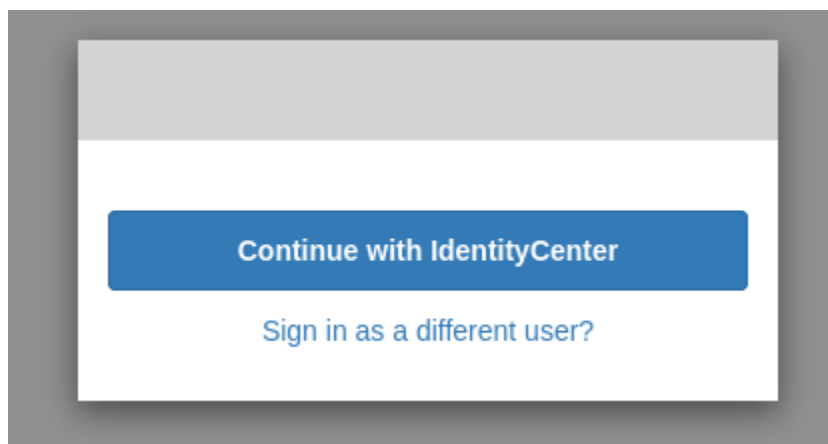
2. En la interfaz de usuario alojada, selecciona Editar
3. En Proveedores de identidad IdentityCenter, selecciona también.
4. Selecciona Save changes (Guardar cambios)

Valida tu configuración

1. A continuación, validaremos la configuración que acabamos de crear iniciando sesión en la PCUI. Inicia sesión en tu portal de la PCUI y ahora verás una opción para iniciar sesión con tu ID corporativa:

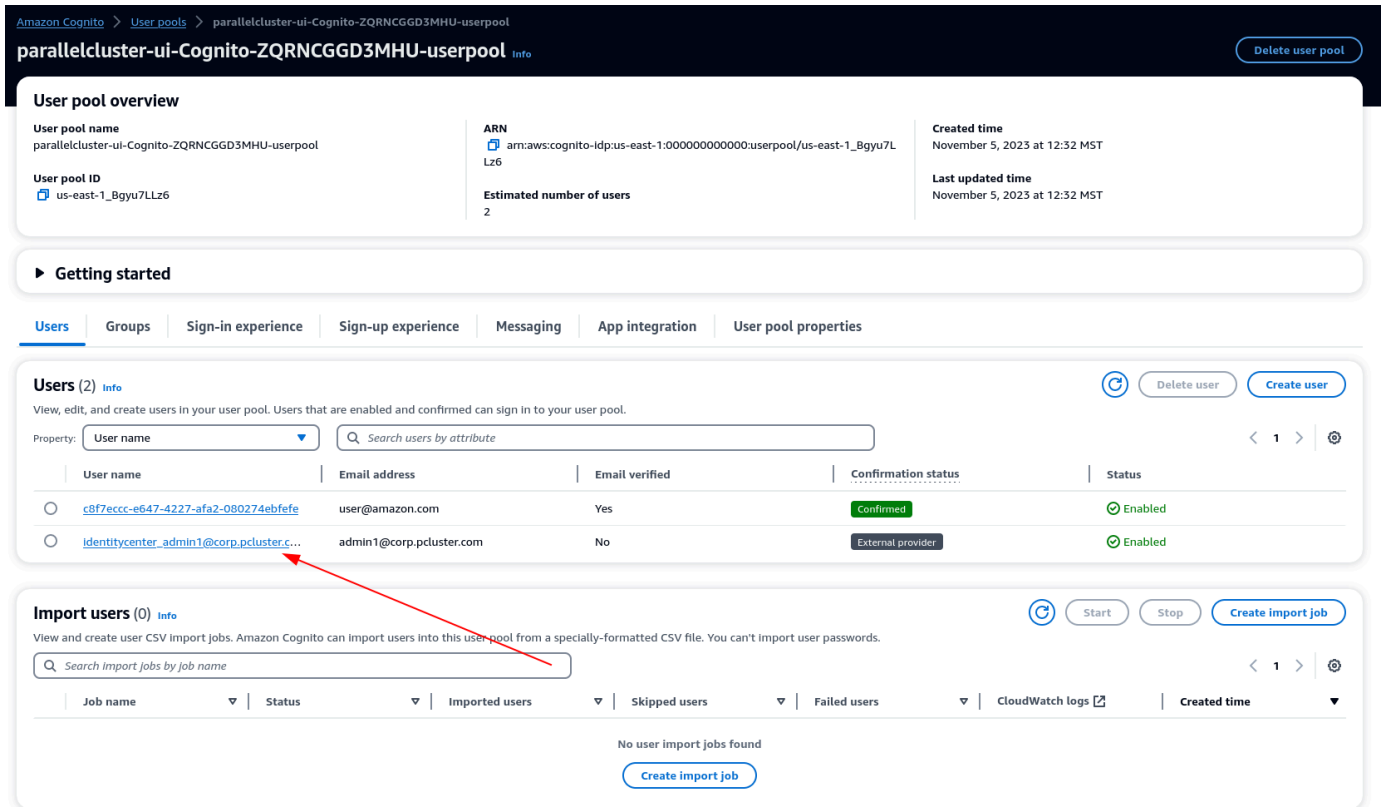


2. Al hacer clic en el IdentityCenter botón, accederá al inicio de sesión del IdP del IAM Identity Center, seguido de una página con sus aplicaciones que incluye la PCUI. Abra esa aplicación.
3. Cuando llegues a la siguiente pantalla, tu usuario se habrá agregado al grupo de usuarios de Cognito.



Convierte a tu usuario en administrador

1. Ahora vaya a la consola Amazon Cognito > Grupos de usuarios y seleccione el usuario recién creado, que debe tener el prefijo identitycenter.



The screenshot shows the AWS Cognito console interface for a user pool named 'parallelcluster-ui-Cognito-ZQRNCGGD3MHU-userpool'. The 'Users' section is active, displaying a table of users. A red arrow points to the user 'identitycenter_admin1@corp.pcluster.c...'.

User pool overview

User pool name parallelcluster-ui-Cognito-ZQRNCGGD3MHU-userpool	ARN arn:aws:cognito-idp:us-east-1:000000000000:userpool/us-east-1_Bgyu7Lz6	Created time November 5, 2023 at 12:32 MST
User pool ID us-east-1_Bgyu7LLz6	Estimated number of users 2	Last updated time November 5, 2023 at 12:32 MST

Getting started

Users | Groups | Sign-in experience | Sign-up experience | Messaging | App integration | User pool properties

Users (2)

View, edit, and create users in your user pool. Users that are enabled and confirmed can sign in to your user pool.

Property: **User name** Search users by attribute

User name	Email address	Email verified	Confirmation status	Status
c8f7eccc-e647-4227-afa2-080274ebfefe	user@amazon.com	Yes	Confirmed	Enabled
identitycenter_admin1@corp.pcluster.c...	admin1@corp.pcluster.com	No	External provider	Enabled

Import users (0)

View and create user CSV import jobs. Amazon Cognito can import users into this user pool from a specially-formatted CSV file. You can't import user passwords.

Search import jobs by job name

Job name	Status	Imported users	Skipped users	Failed users	CloudWatch logs	Created time
No user import jobs found						

2. En Pertenencias a grupos, selecciona Añadir usuario al grupo, selecciona administrador y haz clic en Añadir.
3. Ahora, cuando hagas clic en Continuar, IdentityCenter accederás a la página de la AWS ParallelCluster interfaz de usuario.

AWS ParallelCluster solución de problemas

La AWS ParallelCluster comunidad mantiene una página wiki que proporciona muchos consejos para la solución de problemas en la [AWS ParallelCluster GitHub wiki](#). Para obtener una lista de problemas conocidos, consulte [Problemas conocidos](#).

Temas

- [Intentando crear un clúster](#)
- [Intentando ejecutar un trabajo](#)
- [Intentando actualizar un clúster](#)
- [¿Estás intentando acceder al almacenamiento](#)
- [¿Está intentando eliminar un clúster](#)
- [¿Intentando actualizar la pila AWS ParallelCluster de API](#)
- [Visualización de errores en las inicializaciones de los nodos de computación](#)
- [Solución de problemas de estado del clúster](#)
- [Solución de problemas de implementación del clúster](#)
- [Solución de problemas de escalar](#)
- [Problemas con los grupos de ubicación y el lanzamiento de instancias](#)
- [Directorios que no se pueden reemplazar](#)
- [Solución de problemas de NICE DCV](#)
- [Solución de problemas en clústeres con AWS Batch integración](#)
- [Solución de problemas de integración multiusuario con Active Directory](#)
- [Solución de problemas con las AMI de](#)
- [Solución de problemas cuando se agota el tiempo de espera de una actualización del clúster cuando no se está ejecutando cfn-hup](#)
- [Solución de problemas de redes en bastidor](#)
- [No se pudo actualizar el clúster al realizar una acción onNodeUpdated personalizada](#)
- [Visualización de errores en la configuración personalizada de Slurm](#)
- [Alarmas de clúster](#)
- [Compatibilidad adicional](#)

Intentando crear un clúster

Si utiliza la AWS ParallelCluster versión 3.5.0 y versiones posteriores para crear un clúster y se produce un error en la creación de un clúster con el `--rollback-on-failure` valor establecido en `false`, utilice el comando [`pcluster describe-cluster`](#) CLI para obtener información sobre el estado y el error. En este caso, lo que se espera `clusterStatus` del `pcluster describe-cluster` resultado es `CREATE_FAILED`. Compruebe la `failures` sección de la salida para encontrar el `failureCode` y `failureReason`. Luego, en la siguiente sección, busque la solución adecuada `failureCode` para obtener ayuda adicional sobre la solución de problemas. Para obtener más información, consulte [`pcluster describe-cluster`](#).

En las siguientes secciones, le recomendamos que compruebe los registros del nodo principal, como los `/var/log/chef-client.log` archivos `/var/log/cfn-init.log` and. Para obtener más información sobre AWS ParallelCluster los registros y cómo verlos, consulte [Registros clave para la depuración](#) y [Recuperación y conservación de registros](#).

Si no tiene un `failureCode`, vaya a la AWS CloudFormation consola para ver la pila de clústeres. Compruebe si hay errores en otros recursos para obtener información adicional sobre los errores. `Status Reason HeadNodeWaitCondition` Para obtener más información, consulte [Vea AWS CloudFormation los eventos en CREATE_FAILED](#). Compruebe los `/var/log/chef-client.log` archivos `/var/log/cfn-init.log` y del nodo principal.

failureCode es OnNodeConfiguredExecutionFailure

- ¿Por qué falló?

Proporcionó un script personalizado en la sección `OnNodeConfigured` del nodo principal de la configuración para crear un clúster. Sin embargo, el script personalizado no se pudo ejecutar.

- ¿Cómo resolverlo?

Consulte el `/var/log/cfn-init.log` archivo para obtener más información sobre el error y cómo solucionar el problema en su script personalizado. Cerca del final de este registro, es posible que veas información de ejecución relacionada con el `OnNodeConfigured` script después del `Running command runpostinstall` mensaje.

failureCode es OnNodeConfiguredDownloadFailure

- ¿Por qué falló?

Proporcionó un script personalizado en la sección `OnNodeConfigured` del nodo principal de la configuración para crear un clúster. Sin embargo, no se pudo descargar el script personalizado.

- ¿Cómo resolverlo?

Asegúrese de que la URL sea válida y de que el acceso esté configurado correctamente. Para obtener más información sobre la configuración de los scripts de arranque personalizados, consulte [Acciones de arranque personalizadas](#).

Compruebe los archivos en `/var/log/cfn-init.log`. Al final de este registro, es posible que, después del `Running command runpostinstall` mensaje, aparezca información sobre la ejecución relacionada con el procesamiento de los `OnNodeConfigured` scripts, incluida la descarga.

failureCode es OnNodeConfiguredFailure

- ¿Por qué falló?

Proporcionó un script personalizado en la sección `OnNodeConfigured` del nodo principal de la configuración para crear un clúster. Sin embargo, el uso del script personalizado falló en la implementación del clúster. No se puede determinar una causa inmediata y es necesaria una investigación adicional.

- ¿Cómo resolverlo?

Compruebe los archivos en `/var/log/cfn-init.log`. Cerca del final de este registro, es posible que vea información de ejecución relacionada con el procesamiento de `OnNodeConfigured` scripts después del `Running command runpostinstall` mensaje.

failureCode es OnNodeStartExecutionFailure

- ¿Por qué falló?

Proporcionó un script personalizado en la sección `OnNodeStart` del nodo principal de la configuración para crear un clúster. Sin embargo, el script personalizado no se pudo ejecutar.

- ¿Cómo resolverlo?

Consulte el `/var/log/cfn-init.log` archivo para obtener más información sobre el error y cómo solucionar el problema en su script personalizado. Cerca del final de este registro, es posible que veas información de ejecución relacionada con el `OnNodeStart` script después del `Running command runpreinstall` mensaje.

failureCode es OnNodeStartDownloadFailure

- ¿Por qué falló?

Proporcionó un script personalizado en la sección `OnNodeStart` del nodo principal de la configuración para crear un clúster. Sin embargo, no se pudo descargar el script personalizado.

- ¿Cómo resolverlo?

Asegúrese de que la URL sea válida y de que el acceso esté configurado correctamente. Para obtener más información sobre la configuración de los scripts de arranque personalizados, consulte [Acciones de arranque personalizadas](#).

Compruebe los archivos en `/var/log/cfn-init.log`. Al final de este registro, es posible que, después del `Running command runpreinstall` mensaje, aparezca información sobre la ejecución relacionada con el procesamiento de los `OnNodeStart` scripts, incluida la descarga.

failureCode es OnNodeStartFailure

- ¿Por qué falló?

Proporcionó un script personalizado en la sección `OnNodeStart` del nodo principal de la configuración para crear un clúster. Sin embargo, el uso del script personalizado falló en la implementación del clúster. No se puede determinar una causa inmediata y es necesaria una investigación adicional.

- ¿Cómo resolverlo?

Compruebe los archivos en `/var/log/cfn-init.log`. Cerca del final de este registro, es posible que vea información de ejecución relacionada con el procesamiento de `OnNodeStart` scripts después del `Running command runpreinstall` mensaje.

failureCode es EbsMountFailure

- ¿Por qué falló?

No se pudo montar el volumen de EBS definido en la configuración del clúster.

- ¿Cómo resolverlo?

Consulte el archivo `/var/log/chef-client.log` para conocer los detalles del error.

failureCode es EfsMountFailure

- ¿Por qué falló?

No se pudo montar el volumen de Amazon EFS definido en la configuración del clúster.

- ¿Cómo resolverlo?

Si ha definido un sistema de archivos Amazon EFS existente, asegúrese de que se permita el tráfico entre el clúster y el sistema de archivos. Para obtener más información, consulte [SharedStorage.EfsSettings.FileSystemId](#).

Consulte el archivo `/var/log/chef-client.log` para conocer los detalles del error.

failureCode es FsxMountFailure

- ¿Por qué falló?

No se pudo montar el sistema de archivos Amazon FSx definido en la configuración del clúster.

- ¿Cómo resolverlo?

Si ha definido un sistema de archivos Amazon FSx existente, asegúrese de que se permita el tráfico entre el clúster y el sistema de archivos. Para obtener más información, consulte [SharedStorage.FsxLustreSettings.FileSystemId](#).

Consulte el archivo `/var/log/chef-client.log` para conocer los detalles del error.

failureCode es RaidMountFailure

- ¿Por qué falló?

No se pudieron montar los volúmenes RAID definidos en la configuración del clúster.

- ¿Cómo resolverlo?

Consulte el archivo `/var/log/chef-client.log` para conocer los detalles del error.

failureCode es AmiVersionMismatch

- ¿Por qué falló?

La AWS ParallelCluster versión utilizada para crear la AMI personalizada es diferente de la AWS ParallelCluster versión utilizada para configurar el clúster. En la CloudFormation consola, consulte los detalles de la CloudFormation Status Reason pila de clústeres y compruebe si `HeadNodeWaitCondition` desea obtener información adicional sobre las AWS ParallelCluster versiones y la AMI. Para obtener más información, consulte [Vea AWS CloudFormation los eventos en CREATE_FAILED](#).

- ¿Cómo resolverlo?

Asegúrese de que la AWS ParallelCluster versión utilizada para crear la AMI personalizada sea la misma AWS ParallelCluster que se utilizó para configurar el clúster. Puede cambiar la versión de la AMI personalizada o la versión de la `pcluster` CLI para que sean iguales.

failureCode es InvalidAmi

- ¿Por qué falló?

La AMI personalizada no es válida porque no se creó con AWS ParallelCluster.

- ¿Cómo resolverlo?

Use el `pcluster build-image` comando para crear una AMI haciendo que su AMI sea la imagen principal. Para obtener más información, consulte [pcluster build-image](#).

failureCode está **HeadNodeBootstrapFailure** con **failureReason**

No se pudo configurar el nodo principal.

- ¿Por qué falló?

No se puede determinar una causa inmediata y es necesaria una investigación adicional. Por ejemplo, podría ser que el clúster esté en estado protegido y esto podría deberse a un fallo en el aprovisionamiento de la flota de computación estática.

- ¿Cómo resolverlo?

Consulte el archivo `/var/log/chef-client.log` para conocer los detalles del error.

Note

Si ve la excepción de `RuntimeError Cluster state has been set to PROTECTED mode due to failures detected in static node provisioning`, el clúster está en estado protegido. Para obtener más información, consulte [Cómo depurar el modo protegido](#).

failureCode está **HeadNodeBootstrapFailure** agotando el tiempo de espera para la creación del **failureReason** clúster.

- ¿Por qué falló?

De forma predeterminada, hay un límite de 30 minutos para que se complete la creación del clúster. Si la creación del clúster no se ha completado dentro de este período de tiempo, se produce un error de tiempo de espera. La creación del clúster puede agotarse por diferentes motivos. Por ejemplo, los errores de tiempo de espera pueden deberse a un error en la creación del nodo principal, a un problema de red, a scripts personalizados que tardan demasiado en ejecutarse en el nodo principal, a un error en un script personalizado que se ejecuta en los nodos de procesamiento o a tiempos de espera prolongados para el aprovisionamiento del nodo de procesamiento. No se puede determinar una causa inmediata y es necesaria una investigación adicional.

- ¿Cómo resolverlo?

Consulte los archivos `/var/log/cfn-init.log` y `/var/log/chef-client.log` para conocer los detalles del error. Para obtener más información sobre los registros de AWS ParallelCluster y cómo obtenerlos, consulte [Registros clave para la depuración](#) y [Recuperación y conservación de registros](#).

Puede que descubra lo siguiente en estos registros.

- Visualización de **Waiting for static fleet capacity provisioning** cerca del final del `chef-client.log`

Esto indica que se agotó el tiempo de espera para la creación del clúster al esperar a que se enciendan los nodos estáticos. Para obtener más información, consulte [Visualización de errores en las inicializaciones de los nodos de computación](#).

- La visualización del script del nodo principal de **OnNodeConfigured** o **OnNodeStart** no ha finalizado al final del `cfn-init.log`

Esto indica que el script `OnNodeConfigured` o el script `OnNodeStart` personalizado tardaron mucho en ejecutarse y provocaron un error de tiempo de espera. Compruebe si su script personalizado tiene problemas que puedan provocar que se ejecute durante mucho tiempo. Si el script personalizado tarda mucho en ejecutarse, considere la posibilidad de cambiar el límite de tiempo de espera añadiendo una `DevSettings` sección al archivo de configuración del clúster, como se muestra en el siguiente ejemplo:

```
DevSettings:
  Timeouts:
    HeadNodeBootstrapTimeout: 1800 # default setting: 1800 seconds
```

- No se encuentran los registros o el nodo principal no se creó correctamente

Es posible que el nodo principal no se haya creado correctamente y que no se puedan encontrar los registros. En la CloudFormation consola, consulte los detalles de la pila de clústeres para comprobar si hay más detalles sobre el error.

failureCode está **HeadNodeBootstrapFailure** con **failureReason**

No se pudo iniciar el nodo principal.

- ¿Por qué falló?

No se puede determinar una causa inmediata y es necesaria una investigación adicional.

- ¿Cómo resolverlo?

Compruebe los campos `/var/log/cfn-init.log` y `/var/log/chef-client.log`.

failureCode es ResourceCreationFailure

- ¿Por qué falló?

La creación de algunos recursos falló durante el proceso de creación del clúster. El fallo puede producirse por varias razones: Por ejemplo, los errores en la creación de recursos pueden deberse a problemas de capacidad o a una política de IAM mal configurada.

- ¿Cómo resolverlo?

En la CloudFormation consola, consulte la pila de clústeres para comprobar si hay más detalles sobre el error de creación de recursos.

failureCode es ClusterCreationFailure

- ¿Por qué falló?

No se puede determinar una causa inmediata y es necesaria una investigación adicional.

- ¿Cómo resolverlo?

En la CloudFormation consola, visualice la pila de clústeres y compruebe si hay más detalles sobre el `HeadNodeWaitCondition error. Status Reason`

Compruebe los campos `/var/log/cfn-init.log` y `/var/log/chef-client.log`.

¿Está viendo `WaitCondition timed out...` en la CloudFormation pila

Para obtener más información, consulte [failureCode está HeadNodeBootstrapFailure agotando el tiempo de espera para la creación del failureReason clúster.](#)

Ver `Resource creation cancelled` en CloudFormation pila

Para obtener más información, consulte [failureCode es ResourceCreationFailure.](#)

Failed to run cfn-init...¿Ve u otros errores en la AWS CloudFormation pila

Compruebe los detalles adicionales del fallo `/var/log/cfn-init.log` y `/var/log/chef-client.log` compruebe si hay más detalles.

Visualización de cómo `chef-client.log` termina con **INFO: Waiting for static fleet capacity provisioning**

Esto está relacionado con el tiempo de espera para la creación del clúster cuando se espera a que se enciendan los nodos estáticos. Para obtener más información, consulte [Visualización de errores en las inicializaciones de los nodos de computación](#).

Visualización de **Failed to run preinstall or postinstall in cfn-init.log**

Tiene un `OnNodeStart` script `OnNodeConfigured` or en la `HeadNode` sección de configuración del clúster. El script no funciona correctamente. Compruebe el `/var/log/cfn-init.log` archivo para ver los detalles de error del script personalizado.

¿Está viendo **This AMI was created with xxx, but is trying to be used with xxx...** en la CloudFormation pila

Para obtener más información, consulte [failureCode es AmiVersionMismatch](#).

Ver **This AMI was not baked by AWS ParallelCluster...** en CloudFormation pila

Para obtener más información, consulte [failureCode es InvalidAmi](#).

Visualización de cómo el comando `pcluster create-cluster` no se ejecuta localmente

Consulte el `~/parallelcluster/pcluster-cli.log` en su sistema de archivos local para conocer los detalles del error.

Compatibilidad adicional

Siga las instrucciones de solución de problemas que se indican en [Solución de problemas de implementación del clúster](#).

Comprueba si tu situación está incluida en la sección [Problemas GitHub conocidos](#), en la parte AWS ParallelCluster superior GitHub.

Para obtener asistencia adicional, consulte [Compatibilidad adicional](#).

Intentando ejecutar un trabajo

srun el trabajo interactivo falla con un error **srun: error: fwd_tree_thread: can't find address for <host>, check slurm.conf**

- ¿Por qué falló?

Ejecutaste el `srun` comando para enviar un trabajo y, a continuación, aumentaste el tamaño de la cola utilizando el `pcluster update-cluster` comando sin reiniciar los Slurm daemons una vez finalizada la actualización.

Slurm organiza los daemons de Slurm en una jerarquía de árbol para optimizar la comunicación. Esta jerarquía solo se actualiza cuando se inician los daemons.

Supongamos que se inicia un trabajo y, `srun` a continuación, se ejecuta el `pcluster update-cluster` comando para aumentar el tamaño de la cola. Como parte de la actualización, se lanzan nuevos nodos de cómputo. A continuación, Slurm coloca el trabajo en cola en uno de los nuevos nodos de cómputo. En este caso, tanto los daemons de Slurm como `srun` no detectan los nuevos nodos de computación. `srun` devuelve un error porque no detecta los nuevos nodos.

- ¿Cómo resolverlo?

Reinicia los daemons de Slurm en todos los nodos de procesamiento y use `srun` para enviar su trabajo. Para programar el reinicio de los Slurm daemons, ejecute el `scontrol reboot` comando que reinicia los nodos de procesamiento. Para obtener más información, consulte [Paquetes de conformidad](#) en la documentación de Slurm. También puede reiniciar manualmente los daemons de Slurm de los nodos de computación solicitando el reinicio de los servicios de `systemd` correspondientes.

Job está atascado en el CF estado con **squeue** el comando

Esto podría deberse a que los nodos dinámicos se están encendiendo. Para obtener más información, consulte [Visualización de errores en las inicializaciones de los nodos de computación](#).

Ejecución de trabajos a gran escala y visualización de **nfsd: too many open connections, consider increasing the number of threads in /var/log/messages**

Con un sistema de archivos en red, cuando se alcanzan los límites de la red, el tiempo de espera de E/S también aumenta. Esto puede provocar bloqueos temporales, ya que la red se utiliza para escribir datos para las métricas de red y de E/S.

En el caso de las instancias de quinta generación, utilizamos el controlador ENA para exponer los contadores de paquetes. Estos contadores cuentan los paquetes en función del AWS momento en que la red alcanza los límites de ancho de banda de la instancia. Puede consultar estos contadores para ver si son mayores que 0. Si lo son, significa que ha superado los límites de ancho de banda. Puede ver estos contadores corriendo `ethtool -S eth0 | grep exceeded`.

Superar los límites de la red suele deberse a que se admiten demasiadas conexiones NFS. Esta es una de las primeras cosas que hay que comprobar cuando se alcanzan o se superan los límites de la red.

Por ejemplo, el siguiente resultado muestra los paquetes descartados:

```
$ ethtool -S eth0 | grep exceeded
bw_in_allowance_exceeded: 38750610
bw_out_allowance_exceeded: 1165693
pps_allowance_exceeded: 103
conntrack_allowance_exceeded: 0
linklocal_allowance_exceeded: 0
```

Para evitar recibir este mensaje, considere la posibilidad de cambiar el tipo de instancia del nodo principal por un tipo de instancia con más rendimiento. Considere la posibilidad de trasladar el almacenamiento de datos a sistemas de archivos de almacenamiento compartido que no se exporten como un recurso compartido de NFS, como Amazon EFS o Amazon FSx. Para obtener más información, consulte [Almacenamiento compartido](#) las [mejores prácticas](#) en la AWS ParallelCluster wiki sobre GitHub.

Ejecución de trabajos de MPI

Cómo habilitar el modo de depuración

Para habilitar el modo de depuración de OpenMPI, [consulte ¿Qué controles tiene Open MPI que ayudan a depurar?](#)

[Para habilitar el modo de depuración de IntelMPI, consulte Otras variables de entorno.](#)

Visualización de **MPI_ERRORS_ARE_FATAL** y **OPAL ERROR** en el resultado del trabajo

Estos códigos de error provienen de la capa MPI de su aplicación. Para obtener información sobre cómo obtener los registros de depuración de MPI de su aplicación, consulte. [Cómo habilitar el modo de depuración](#)

Una posible causa de este error es que la aplicación se ha compilado para una implementación de MPI específica, como OpenMPI, y está intentando ejecutarla con una implementación de MPI diferente, como IntelMPI. Asegúrese de compilar y ejecutar la aplicación con la misma implementación de MPI.

Se utiliza **mpirun** con el DNS administrado desactivado

[En el caso de los clústeres creados con SlurmSettings/Dns/DisableManagedDns y UseEc2Hostnames configurados en true, el DNS no resuelve el nombre del Slurm nodo.](#)

Slurmpuede iniciar procesos de MPI cuando nodenames no están habilitados y si el trabajo de MPI se ejecuta en un contexto. Slurm Recomendamos seguir las instrucciones de la Guía del [usuario de Slurm MPI para ejecutar trabajos de MPI](#) con él. Slurm

Intentando actualizar un clúster

pcluster update-clusterel comando no se ejecuta localmente

Consulte el `~/parallelcluster/pcluster-cli.log` en su sistema de archivos local para conocer los detalles del error.

Visualización de **clusterStatus** como **UPDATE_FAILED** con el comando **pcluster describe-cluster**

Si la actualización de la pila de clústeres se ha revertido, compruebe los detalles del error en el `/var/log/chef-client.logs` archivo.

Comprueba si tu problema aparece en la sección [Problemas GitHub conocidos, en la sección correspondiente](#). AWS ParallelCluster GitHub

Se agotó el tiempo de espera de la actualización del clúster

Podría deberse a un problema relacionado con la falta de `cf-n-hup` ejecución. Si el daemon de `cf-n-hup` se elimina por una causa externa, no se reinicia automáticamente. Si `cf-n-hup` no se está ejecutando, durante una actualización del clúster, la CloudFormation pila inicia el proceso de actualización según lo previsto, pero el procedimiento de actualización no se activa en el nodo principal y, finalmente, se agota el tiempo de espera para el despliegue de la pila. Para obtener más información, consulte [Solución de problemas cuando se agota el tiempo de espera de una actualización del clúster cuando no se está ejecutando `cf-n-hup`](#) para solucionar el problema y recuperarse de él.

¿Estás intentando acceder al almacenamiento

Uso de un sistema de archivos de Amazon FSx para Lustre externo

Asegúrese de que se permita el tráfico entre el clúster y el sistema de archivos. Si se usa un sistema de archivos ya existente, debe asociarse a un grupo de seguridad que permita el tráfico TCP de entrada y salida desde a través del puerto . Para obtener más información sobre cómo configurar los grupos de seguridad, consulte [FileSystemId](#).

Uso de un sistema de archivos externo de Amazon Elastic File System

Asegúrese de que se permita el tráfico entre el clúster y el sistema de archivos. Si se usa un sistema de archivos ya existente, debe asociarse a un grupo de seguridad que permita el tráfico TCP de entrada y salida desde a través del puerto . Para obtener más información sobre cómo configurar grupos de seguridad, consulte [FileSystemId](#).

¿Está intentando eliminar un clúster

El `pcluster delete-cluster` comando no se puede ejecutar localmente

Compruebe el `~/parallelcluster/pcluster-cli.log` archivo en su sistema de archivos local.

La pila de clústeres no se puede eliminar

Si la pila de clústeres no se elimina, compruebe el mensaje de eventos de la CloudFormation pila.

Comprueba si tu problema aparece en la sección [Problemas GitHub conocidos](#), AWS ParallelCluster en la sección correspondiente GitHub.

¿Intentando actualizar la pila AWS ParallelCluster de API

Comprueba si tu problema aparece en la sección [Problemas GitHub conocidos](#), AWS ParallelCluster en la sección correspondiente GitHub.

Visualización de errores en las inicializaciones de los nodos de computación

Visualización de **Node bootstrap error** en **clustermgtd.log**

El problema está relacionado con la falla del arranque de los nodos de cómputo. Para obtener información sobre cómo depurar un problema relacionado con el modo protegido de un clúster, consulte. [Cómo depurar el modo protegido](#)

Si configuré reservas de capacidad (ODCR) bajo demanda o instancias reservadas zonales

ODCR que incluyen instancias que tienen varias interfaces de red, como P4d, P4de y Trainium (Trn) AWS

En el archivo de configuración del clúster, compruebe que HeadNode se encuentre en una subred pública y que los nodos de procesamiento estén en una subred privada.

Los ODCR son ODCRS de destino

Visualización de **Unable to read file '/opt/slurm/etc/pcluster/run_instances_overrides.json'**. a pesar de que ya he implementado **/opt/slurm/etc/pcluster/run_instances_overrides.json** siguiendo las instrucciones que dadas en [Inicio de instancias con ODCR \(reservas de capacidad bajo demanda\)](#)

Si utilizas AWS ParallelCluster las versiones 3.1.1 a 3.2.1 con los ODCR de destino y también utilizas el archivo JSON de [anulación de instancias ejecutadas, es posible que el archivo JSON](#) no tenga el formato correcto. Es posible que aparezca un error en `clustermgtd.log`, por ejemplo, el siguiente:

```
Unable to read file '/opt/slurm/etc/pcluster/run_instances_overrides.json'.
Using default: {} in /var/log/parallelcluster/clustermgtd.
```

Compruebe que el formato del archivo JSON es correcto ejecutando lo siguiente:

```
$ echo /opt/slurm/etc/pcluster/run_instances_overrides.json | jq
```

Visualización de **Found RunInstances parameters override.** en `clustermgtd.log` cuando falló la creación del clúster o en `slurm_resume.log` cuando falló la tarea de ejecución

Si utiliza [instancias de ejecución que anulan el archivo JSON](#), compruebe que ha establecido correctamente el nombre de la cola y el nombre de los recursos de cómputo en el archivo `/opt/slurm/etc/pcluster/run_instances_overrides.json`.

Visualización de **An error occurred (InsufficientInstanceCapacity)** en `slurm_resume.log` cuando no puedo ejecutar un trabajo o en `clustermgtd.log` cuándo no puedo crear un clúster

Uso de PG-ODCR (grupo de ubicación ODCR)

Al crear un ODCR con un grupo de ubicación asociado, se debe utilizar el mismo nombre de grupo de ubicación en el archivo de configuración. Establezca el [nombre del grupo de ubicación](#) correspondiente en la configuración del clúster.

Uso de instancias reservadas

Si utiliza instancias reservadas zonales con `PlacementGroup/Enabled` a `true` en la configuración del clúster, es posible que aparezca un error como el siguiente:

We currently do not have sufficient `trn1.32xlarge` capacity in the Availability Zone you requested (`us-east-1d`). Our system will be working on provisioning additional capacity.

You can currently get `trn1.32xlarge` capacity by not specifying an Availability Zone in your request or choosing `us-east-1a`, `us-east-1b`, `us-east-1c`, `us-east-1e`, `us-east-1f`.

Es posible que esto se deba a que las instancias reservadas zonales no están ubicadas en la misma UC (o columna vertebral), lo que puede provocar errores de capacidad insuficiente (ICE) al utilizar grupos de ubicación. Para comprobar este caso, inhabilite la configuración de `PlacementGroup` grupo en la configuración del clúster para determinar si el clúster puede asignar las instancias.

Visualización de **An error occurred (VcpuLimitExceeded)** en `slurm_resume.log` cuando no puedo ejecutar un trabajo o en `clustermgtd.log` cuando no puedo crear un clúster

Compruebe los límites de vCPU de su cuenta para el tipo de instancia EC2 específico que esté utilizando. Si ve cero o menos vCPU de las que solicita, solicite un aumento de sus límites. Para obtener información sobre cómo ver los límites actuales y solicitar nuevos límites, consulte las [cuotas de servicio de Amazon EC2](#) en la Guía del usuario de Amazon EC2.

Visualización de **An error occurred (InsufficientInstanceCapacity)** en `slurm_resume.log` cuando no puedo ejecutar un trabajo o en `clustermgtd.log` cuando no puedo crear un clúster

Tiene un problema de capacidad insuficiente. Siga <https://aws.amazon.com/premiumsupport/knowledge-center/ec2-insufficient-capacity-errors/> para solucionar el problema.

Visualización de los nodos que están en estado **DOWN** con **Reason (Code:InsufficientInstanceCapacity)...**

Tiene un problema de capacidad insuficiente. Siga <https://aws.amazon.com/premiumsupport/knowledge-center/ec2-insufficient-capacity-errors/> para solucionar el problema. Para obtener más información sobre AWS ParallelCluster el modo de conmutación por error rápida y con capacidad insuficiente, consulte. [Conmutación por error rápida de capacidad insuficiente en el clúster de Slurm](#)

Visualización de **cannot change locale (en_US.utf-8) because it has an invalid name** en **slurm_resume.log**

Esto puede ocurrir si el proceso de yum instalación no se ha realizado correctamente y ha dejado la configuración regional en un estado incoherente. Por ejemplo, esto puede producirse cuando un usuario finaliza el proceso de instalación.

Para verificar la causa, realice las siguientes acciones:

- Ejecute su `sudo pcluster-admin`.

El intérprete de comandos muestra un error, como `cannot change locale...no such file or directory`.

- Ejecute `localedef --list`.

Devuelve una lista vacía o no contiene la configuración regional predeterminada.

- Marque el último yum comando con `yum history` y `yum history info #ID` ¿La última identificación tiene `Return-Code: Success`?

Si el último identificador no lo tiene `Return-Code: Success`, es posible que los scripts posteriores a la instalación no se hayan ejecutado correctamente.

Para solucionar el problema, intenta volver a crear la configuración regional con `yum reinstall glibc-all-langpacks` Tras la reconstrucción, su `sudo pcluster-admin` no muestra ningún error o advertencia si el problema se ha solucionado.

Ninguno de los escenarios anteriores se aplica a mi situación

Para solucionar problemas de inicialización de los nodos de procesamiento, consulte [Solución de problemas de inicialización de nodos](#)

Compruebe si su situación está incluida en la sección [Problemas GitHub conocidos](#), en la sección correspondiente. AWS ParallelCluster GitHub

Para obtener asistencia adicional, consulte [Compatibilidad adicional](#).

Solución de problemas de estado del clúster

Las métricas del estado del clúster se añaden al panel de Amazon CloudWatch AWS ParallelCluster a partir de AWS ParallelCluster versión 3.6.0. En las siguientes secciones, puede obtener información sobre las métricas de estado del panel y las acciones que puede seguir para solucionar y solucionar problemas.

Temas

- [Consulta el gráfico de errores de aprovisionamiento de instancias](#)
- [Ver el gráfico de errores de instancias en mal estado](#)
- [Consulte el gráfico de tiempo de inactividad de Compute Fleet](#)

Consulta el gráfico de errores de aprovisionamiento de instancias

Si ves un valor distinto de cero en el Instance Provisioning Errors gráfico, significa que la instancia EC2 que respalda los nodos slurm no se pudo lanzar en la API CreateFleet o RunInstance.

¿Está viendo **IAMPolicyErrors**

- ¿Qué ha pasado?

No se pudieron iniciar varias instancias, lo que se debió a que los permisos eran insuficientes y el código de error era insuficiente UnauthorizedOperation.

- ¿Cómo resolverlo?

Si ha configurado una personalizada [InstanceRole](#) o [InstanceProfile](#), compruebe sus políticas de IAM y compruebe que está utilizando las credenciales correctas.

Compruebe el `clustermgtd` archivo para ver los detalles de los errores de los nodos estáticos. Compruebe el `slurm_resume.log` archivo para ver los detalles de los errores de los nodos dinámicos. Utilice los detalles para obtener más información sobre los permisos que faltan y que se deben añadir.

¿Está viendo **VcpuLimitErrors**

- ¿Qué ha pasado?

AWS ParallelCluster no pudo lanzar instancias porque alcanzó el límite de vCPU Cuenta de AWS para un tipo de instancia EC2 específico que configuró para los nodos de procesamiento del clúster.

- ¿Cómo resolverlo?

Compruebe si hay `VcpuLimitExceeded` algún error en el `clustermgtd` archivo para los nodos estáticos y compruebe si hay nodos dinámicos en el `slurm_resume.log` archivo para obtener información adicional. Para resolver este problema, puede solicitar un aumento de los límites de vCPU. Para obtener más información acerca de cómo ver los límites actuales y solicitar nuevos límites, consulte [Cuotas de servicio de Amazon EC2](#) en la Guía del usuario de Amazon EC2 para instancias de Linux.

¿Está viendo **VolumeLimitErrors**

- ¿Qué ha pasado?

Ha alcanzado el límite de volumen de Amazon EBS y AWS ParallelCluster no puede lanzar instancias con un código de error `InsufficientVolumeCapacity` o `VolumeLimitExceeded`. Cuenta de AWS

- ¿Cómo resolverlo?

Compruebe si hay nodos estáticos en el `slurm_resume.log` archivo y si hay nodos dinámicos para obtener detalles adicionales sobre el límite de volumen. `clustermgtd` Para resolver este problema, puede utilizar otro Región de AWS, limpiar los volúmenes existentes o ponerse en contacto con el AWS Support Center para enviar una solicitud para aumentar el límite de volumen de Amazon EBS.

¿Estás viendo **InsufficientCapacityErrors**

- ¿Qué ha pasado?

AWS ParallelCluster no tiene la capacidad suficiente para lanzar instancias EC2 en los nodos posteriores.

- ¿Cómo resolverlo?

Compruebe si hay nodos estáticos en el archivo `clustermgtd` y el archivo `slurm_resume.log` por si hay nodos dinámicos para obtener los detalles del error de capacidad insuficiente. Para solucionar el problema, sigue las instrucciones de <https://aws.amazon.com/premiumsupport/knowledge-center/ec2-insufficient-capacity-errors/>.

OtherInstanceLaunchFailures

- ¿Qué ha pasado?

La instancia EC2 para respaldar los nodos de cómputo no se pudo iniciar con la API `CreateFleet` o `RunInstance`.

- ¿Cómo resolverlo?

Compruebe si hay nodos estáticos en el archivo `clustermgtd` y el archivo `slurm_resume.log` por si hay nodos dinámicos para obtener los detalles del error.

Ver el gráfico de errores de instancias en mal estado

- ¿Qué ha pasado?

Se lanzaron varias instancias de cómputo, pero más tarde se cancelaron por estar en mal estado.

- ¿Cómo resolverlo?

Para obtener más información acerca de la solución de problemas de nodos dañados, consulte [Solución de problemas de sustituciones y terminaciones inesperadas de nodos](#).

¿Viendo **InstanceBootstrapTimeoutError**

- ¿Qué ha pasado?

Una instancia no puede unirse al clúster dentro de `resume_timeout` (para nodos dinámicos) o `node_replacement_timeout` (para nodos estáticos). Esto puede ocurrir si la red no está configurada correctamente para los nodos de cómputo o si los scripts personalizados que se ejecutan en el nodo de cómputo tardan demasiado en finalizar.

- ¿Cómo resolverlo?

En el caso de los nodos dinámicos, compruebe en el `clustermgtd` registro (`/var/log/parallelcluster/clustermgtd`) la dirección IP del nodo de procesamiento y errores como los siguientes:

```
Node bootstrap error: Resume timeout expires for node
```

En el caso de los nodos estáticos, compruebe en el `clustermgtd` registro (`/var/log/parallelcluster/clustermgtd`) la dirección IP del nodo de procesamiento y errores como los siguientes:

```
Node bootstrap error: Replacement timeout expires for node ... in replacement.
```

Para obtener más información, compruebe si hay errores en el `/var/log/cloud-init-output.log` archivo. Puede recuperar las direcciones IP de los nodos de cómputo problemáticos de `clustermgtd` los archivos de `slurm_resume` registro.

¿Viendo **EC2HealthCheckErrors**

- ¿Qué ha pasado?

Una instancia no pasó una comprobación de estado de EC2.

- ¿Cómo resolverlo?

Para obtener información acerca de cómo solucionar este problema, consulta [Solución de problemas de las instancias con comprobaciones de estado no superadas](#).

¿Está viendo **ScheduledEventHealthCheckErrors**

- ¿Qué ha pasado?

Una instancia no pasó la comprobación del estado de un evento programado de EC2 y no está en buen estado.

- ¿Cómo resolverlo?

Para obtener información sobre cómo solucionar este problema, consulta [Eventos programados para sus instancias](#).

¿Está viendo **NoCorrespondingInstanceErrors**

- ¿Qué ha pasado?

AWS ParallelCluster no puede encontrar instancias que respalden los nodos. Es probable que los nodos se hayan autoterminado durante las operaciones de arranque.

[SlurmQueues/CustomActions/OnNodeStart](#) se pueden producir errores de secuencia de [OnNodeConfigured](#) comandos o de red. **NoCorrespondingInstanceErrors**

- ¿Cómo resolverlo?

Para obtener más información, compruebe el nodo `/var/log/cloud-init-output.log` de cómputo.

Consulte el gráfico de tiempo de inactividad de Compute Fleet

Veo un valor **MaxDynamicNodeIdleTime** significativamente más largo que el umbral de reducción del tiempo de inactividad

- ¿Qué ha pasado?

La instancia no está finalizando correctamente. **MaxDynamicNodeIdleTime** muestra el tiempo máximo en segundos que un nodo dinámico, respaldado por una instancia EC2, permanece inactivo. El umbral de reducción del tiempo de inactividad se deriva del parámetro de configuración del clúster [ScaledownIdleTime](#). Cuando un nodo de cómputo ha estado inactivo durante más de unos segundos con la reducción del tiempo de inactividad, se Slurm apaga el nodo y se cierra la instancia de respaldo AWS ParallelCluster. En este caso, algo impide la finalización de la instancia.

- ¿Cómo resolverlo?

Para obtener información acerca de este problema, consulte [Reemplazar, terminar o apagar instancias y nodos problemáticos](#) en [Solución de problemas de escalar](#).

Solución de problemas de implementación del clúster

Si el clúster no se crea y revierte la creación de la pila, puede revisar los archivos de registro para diagnosticar el problema. El mensaje de error debe ser similar al siguiente:

```
$ pcluster create-cluster --cluster-name mycluster --region eu-west-1 \
--cluster-configuration cluster-config.yaml
{
  "cluster": {
    "clusterName": "mycluster",
    "cloudformationStackStatus": "CREATE_IN_PROGRESS",
    "cloudformationStackArn": "arn:aws:cloudformation:eu-west-1:xxx:stack/
mycluster/1bf6e7c0-0f01-11ec-a3b9-024fcc6f3387",
    "region": "eu-west-1",
    "version": "3.7.0",
    "clusterStatus": "CREATE_IN_PROGRESS"
  }
}

$ pcluster describe-cluster --cluster-name mycluster --region eu-west-1
{
  "creationTime": "2021-09-06T11:03:47.696Z",
  ...
  "cloudFormationStackStatus": "ROLLBACK_IN_PROGRESS",
  "clusterName": "mycluster",
  "computeFleetStatus": "UNKNOWN",
  "cloudformationStackArn": "arn:aws:cloudformation:eu-west-1:xxx:stack/
mycluster/1bf6e7c0-0f01-11ec-a3b9-024fcc6f3387",
  "lastUpdatedTime": "2021-09-06T11:03:47.696Z",
  "region": "eu-west-1",
  "clusterStatus": "CREATE_FAILED"
}
```

Temas

- [Vea AWS CloudFormation los eventos en CREATE_FAILED](#)
- [Utilice la CLI para ver los flujos de registro](#)
- [Vuelva a crear el clúster fallido con rollback-on-failure](#)

Vea AWS CloudFormation los eventos en **CREATE_FAILED**

Puede usar la consola o la AWS ParallelCluster CLI para ver CloudFormation los eventos de los CREATE_FAILED errores y ayudar a encontrar la causa raíz.

Temas

- [Vea los eventos en la CloudFormation consola](#)

- [Utilice la CLI para ver y filtrar CloudFormation eventos en CREATE_FAILED](#)

Vea los eventos en la CloudFormation consola

Para obtener más información sobre la causa del "CREATE_FAILED" estado, puedes usar la CloudFormation consola.

Consulta los mensajes de CloudFormation error de la consola.

1. Inicie sesión en AWS Management Console y vaya a <https://console.aws.amazon.com/cloudformation>.
2. Seleccione la pila denominada *cluster_name*.
3. Seleccione la pestaña Eventos.
4. Compruebe el estado del recurso que no se pudo crear desplazándose por la lista de eventos del recurso por identificador lógico. Si no se pudo crear una subtarea, retroceda para encontrar el evento de recurso fallido.
5. Por ejemplo, si ve el siguiente mensaje de estado, debe usar tipos de instancia que no superen el límite de vCPU actual ni solicitar más capacidad de vCPU.

```
2022-02-04 16:09:44 UTC-0800 HeadNode CREATE_FAILED You have requested more vCPU
capacity than your current vCPU limit of 0 allows
    for the instance bucket that the specified instance type belongs to. Please
visit http://aws.amazon.com/contact-us/ec2-request to request an adjustment to
this limit.
    (Service: AmazonEC2; Status Code: 400; Error Code: VcpuLimitExceeded; Request
ID: a9876543-b321-c765-d432-dcba98766789; Proxy: null).
```

Utilice la CLI para ver y filtrar CloudFormation eventos en CREATE_FAILED

Para diagnosticar el problema de creación del clúster, puede usar el [pcluster get-cluster-stack-events](#) comando filtrando por CREATE_FAILED estado. Para obtener más información, consulte [Filtrar los AWS CLI resultados](#) en la Guía del AWS Command Line Interface usuario.

```
$ pcluster get-cluster-stack-events --cluster-name mycluster --region eu-west-1 \
  --query 'events[?resourceStatus==`CREATE_FAILED`]'
[
  {
```

```

    "eventId": "3ccdedd0-0f03-11ec-8c06-02c352fe2ef9",
    "physicalResourceId": "arn:aws:cloudformation:eu-west-1:xxx:stack/
mycluster/1bf6e7c0-0f02-11ec-a3b9-024fcc6f3387",
    "resourceStatus": "CREATE_FAILED",
    "resourceStatusReason": "The following resource(s) failed to create: [HeadNode].
",
    "stackId": "arn:aws:cloudformation:eu-west-1:xxx:stack/
mycluster/1bf6e7c0-0f02-11ec-a3b9-024fcc6f3387",
    "stackName": "mycluster",
    "logicalResourceId": "mycluster",
    "resourceType": "AWS::CloudFormation::Stack",
    "timestamp": "2021-09-06T11:11:51.780Z"
  },
  {
    "eventId": "HeadNode-CREATE_FAILED-2021-09-06T11:11:50.127Z",
    "physicalResourceId": "i-04e91cc1f4ea796fe",
    "resourceStatus": "CREATE_FAILED",
    "resourceStatusReason": "Received FAILURE signal with UniqueId
i-04e91cc1f4ea796fe",
    "resourceProperties": "{\"LaunchTemplate\":{\"Version\":\"1\"},\"LaunchTemplateId
\": \"lt-057d2b1e687f05a62\"}}",
    "stackId": "arn:aws:cloudformation:eu-west-1:xxx:stack/
mycluster/1bf6e7c0-0f02-11ec-a3b9-024fcc6f3387",
    "stackName": "mycluster",
    "logicalResourceId": "HeadNode",
    "resourceType": "AWS::EC2::Instance",
    "timestamp": "2021-09-06T11:11:50.127Z"
  }
]

```

En el ejemplo anterior, el error se produjo en la configuración del nodo principal.

Utilice la CLI para ver los flujos de registro

Para solucionar este tipo de problemas, puede enumerar los flujos de registro disponibles en el nodo principal [pcluster list-cluster-log-streams](#) filtrando `node-type` y analizando el contenido de los flujos de registro.

```

$ pcluster list-cluster-log-streams --cluster-name mycluster --region eu-west-1 \
--filters 'Name=node-type,Values=HeadNode'
{
  "logStreams": [

```

```

{
  "logStreamArn": "arn:aws:logs:eu-west-1:xxx:log-group:/aws/parallelcluster/
mycluster-202109061103:log-stream:ip-10-0-0-13.i-04e91cc1f4ea796fe.cfn-init",
  "logStreamName": "ip-10-0-0-13.i-04e91cc1f4ea796fe.cfn-init",
  ...
},
{
  "logStreamArn": "arn:aws:logs:eu-west-1:xxx:log-group:/aws/parallelcluster/
mycluster-202109061103:log-stream:ip-10-0-0-13.i-04e91cc1f4ea796fe.chef-client",
  "logStreamName": "ip-10-0-0-13.i-04e91cc1f4ea796fe.chef-client",
  ...
},
{
  "logStreamArn": "arn:aws:logs:eu-west-1:xxx:log-group:/aws/parallelcluster/
mycluster-202109061103:log-stream:ip-10-0-0-13.i-04e91cc1f4ea796fe.cloud-init",
  "logStreamName": "ip-10-0-0-13.i-04e91cc1f4ea796fe.cloud-init",
  ...
},
...
]
}

```

Los dos flujos de registro principales que puede utilizar para buscar errores de inicialización son los siguientes:

- `cfn-init` es el registro del script `cfn-init`. En primer lugar, compruebe este flujo de registro. Es probable que veas el Command `chef failed error` en este registro. Consulte las líneas inmediatamente anteriores a esta línea para obtener información más específica relacionada con el mensaje de error. Para obtener más información, consulte [cfn-init](#).
- `cloud-init` es el registro de [cloud-init](#). Si no ve nada en `cfn-init`, intente revisar este registro a continuación.

Para recuperar el contenido del flujo de registro, utilice las siguientes opciones [pcluster get-cluster-log-events](#) (tenga en cuenta la `--limit 5` opción para limitar el número de eventos recuperados):

```

$ pcluster get-cluster-log-events --cluster-name mycluster \
  --region eu-west-1 --log-stream-name ip-10-0-0-13.i-04e91cc1f4ea796fe.cfn-init \
  --limit 5
{
  "nextToken": "f/36370880979637159565202782352491087067973952362220945409/s",

```

```
"prevToken": "b/36370880752972385367337528725601470541902663176996585497/s",
"events": [
  {
    "message": "2021-09-06 11:11:39,049 [ERROR] Unhandled exception during build:
Command runpostinstall failed",
    "timestamp": "2021-09-06T11:11:39.049Z"
  },
  {
    "message": "Traceback (most recent call last):\n File \"/opt/aws/bin/
cfn-init\"", line 176, in <module>\n   worklog.build(metadata, configSets)\n
File \"/usr/lib/python3.7/site-packages/cfnbootstrap/construction.py\"", line
135, in build\n   Contractor(metadata).build(configSets, self)\n File \"/
usr/lib/python3.7/site-packages/cfnbootstrap/construction.py\"", line 561, in
build\n   self.run_config(config, worklog)\n File \"/usr/lib/python3.7/
site-packages/cfnbootstrap/construction.py\"", line 573, in run_config\n
CloudFormationCarpenter(config, self._auth_config).build(worklog)\n File \"/usr/
lib/python3.7/site-packages/cfnbootstrap/construction.py\"", line 273, in build\n
self._config.commands)\n File \"/usr/lib/python3.7/site-packages/cfnbootstrap/
command_tool.py\"", line 127, in apply\n   raise ToolError(u"Command %s failed\" %
name)",
    "timestamp": "2021-09-06T11:11:39.049Z"
  },
  {
    "message": "cfnbootstrap.construction_errors.ToolError: Command runpostinstall
failed",
    "timestamp": "2021-09-06T11:11:39.049Z"
  },
  {
    "message": "2021-09-06 11:11:49,212 [DEBUG] CloudFormation client initialized
with endpoint https://cloudformation.eu-west-1.amazonaws.com",
    "timestamp": "2021-09-06T11:11:49.212Z"
  },
  {
    "message": "2021-09-06 11:11:49,213 [DEBUG] Signaling resource HeadNode in stack
mycluster with unique ID i-04e91cc1f4ea796fe and status FAILURE",
    "timestamp": "2021-09-06T11:11:49.213Z"
  }
]
```

En el ejemplo anterior, el error se debe a un `runpostinstall` error, por lo que está estrictamente relacionado con el contenido del script de arranque personalizado utilizado en el parámetro de `OnNodeConfigured` configuración del [CustomActions](#).

Vuelva a crear el clúster fallido con **rollback-on-failure**

AWS ParallelCluster crea flujos de CloudWatch registro de clústeres en grupos de registros. Puede ver estos registros en los paneles personalizados de la CloudWatch consola o en los grupos de registros. Para obtener más información, consulte [Integración con Amazon CloudWatch Logs](#) y [Panel de Amazon CloudWatch](#). Si no hay flujos de registro disponibles, el error puede deberse al script de arranque [CustomActions](#) personalizado o a un problema relacionado con la AMI. Para diagnosticar el problema de creación en este caso, vuelva a crear el clúster utilizando `pcluster create-cluster`, incluido el `--rollback-on-failure` parámetro establecido en `false`. A continuación, utilice SSH para ver el clúster, tal y como se muestra a continuación:

```
$ pcluster create-cluster --cluster-name mycluster --region eu-west-1 \
  --cluster-configuration cluster-config.yaml --rollback-on-failure false
{
  "cluster": {
    "clusterName": "mycluster",
    "cloudformationStackStatus": "CREATE_IN_PROGRESS",
    "cloudformationStackArn": "arn:aws:cloudformation:eu-west-1:xxx:stack/
mycluster/1bf6e7c0-0f01-11ec-a3b9-024fcc6f3387",
    "region": "eu-west-1",
    "version": "3.7.0",
    "clusterStatus": "CREATE_IN_PROGRESS"
  }
}
$ pcluster ssh --cluster-name mycluster
```

Una vez que haya iniciado sesión en el nodo principal, encontrará tres archivos de registro principales que podrá usar para encontrar el error.

- `/var/log/cfn-init.log` es el registro del script `cfn-init`. Compruebe primero este registro. Es probable que veas un error como `Command chef failed` el de este registro. Consulte las líneas inmediatamente anteriores a esta línea para obtener información más específica relacionada con el mensaje de error. Para obtener más información, consulte [cfn-init](#).
- `/var/log/cloud-init.log` es el registro de [cloud-init](#). Si no ve nada en `cfn-init.log`, intente revisar este registro a continuación.
- `/var/log/cloud-init-output.log` es el resultado de los comandos ejecutados por [cloud-init](#). Esto incluye el resultado de `cfn-init`. En la mayoría de los casos, no es necesario consultar este registro para solucionar problemas de este tipo.

Solución de problemas de escalar

Esta sección es relevante para los clústeres que se instalaron con la AWS ParallelCluster versión 3.0.0 y versiones posteriores con el programador de tareas de Slurm. Para obtener más información acerca de la configuración de colas de consultas, consulte [Configuración de varias colas](#).

Si uno de sus clústeres en ejecución tiene problemas, coloque el clúster en ese STOPPED estado ejecutando el siguiente comando antes de empezar a solucionar el problema. Esto evita incurrir en costes inesperados.

```
$ pcluster update-compute-fleet --cluster-name mycluster \  
  --status STOP_REQUESTED
```

Puede enumerar los flujos de registro disponibles en los nodos del clúster mediante el [pcluster list-cluster-log-streams](#) comando y filtrarlos mediante uno `private-dns-name` de los nodos que fallan o el nodo principal:

```
$ pcluster list-cluster-log-streams --cluster-name mycluster --region eu-west-1 \  
  --filters 'Name=private-dns-name,Values=ip-10-0-0-101'
```

A continuación, puede recuperar el contenido del flujo de registro para analizarlo mediante el [pcluster get-cluster-log-events](#) comando y pasar el `--log-stream-name` correspondiente a uno de los registros clave que se mencionan en la siguiente sección:

```
$ pcluster get-cluster-log-events --cluster-name mycluster \  
  --region eu-west-1 --log-stream-name ip-10-0-0-13.i-04e91cc1f4ea796fe.cfn-init
```

AWS ParallelCluster crea flujos de CloudWatch registro de clústeres en grupos de registros. Puede ver estos registros en los paneles personalizados de la CloudWatch consola o en los grupos de registros. Para obtener más información, consulte [Integración con Amazon CloudWatch Logs](#) y [Panel de Amazon CloudWatch](#).

Temas

- [Registros clave para la depuración](#)
- [Visualización de un error de `InsufficientInstanceCapacity` en `slurm_resume.log` cuando no puedo ejecutar un trabajo o en `clustermgtd.log` cuando no puedo crear un clúster](#)
- [Solución de problemas de inicialización de nodos](#)

- [Solución de problemas de sustituciones y terminaciones inesperadas de nodos](#)
- [Reemplazar, terminar o apagar instancias y nodos problemáticos](#)
- [Estado de la cola \(partición\) Inactive](#)
- [Solución de otros problemas de nodos y trabajos conocidos](#)

Registros clave para la depuración

En la siguiente tabla se proporciona una descripción general de los registros clave del nodo principal:

- `/var/log/cfn-init.log`- Este es el registro de AWS CloudFormation inicio. Contiene todos los comandos que se ejecutaron al configurar una instancia. Se usa para solucionar problemas de inicialización.
- `/var/log/chef-client.log`- Este es el registro del cliente de Chef. Contiene todos los comandos que se ejecutaron a través de Chef/CINC. Se usa para solucionar problemas de inicialización.
- `/var/log/parallelcluster/slurm_resume.log`- Es un ResumeProgram registro. Lanza instancias para nodos dinámicos. Se usa para solucionar problemas de lanzamiento de nodos dinámicos.
- Este es el registro de `clusteradm`. Se llama cuando se terminan las instancias de los nodos dinámicos. Se usa para solucionar problemas de terminación de nodos dinámicos. Cuando revise este registro, también debe comprobar el registro de `clusteradm`.
- Este es el registro de `clusteradm`. Se ejecuta como el daemon centralizado que gestiona la mayoría de las acciones operativas del clúster. Se usa para solucionar cualquier problema de inicio, finalización o funcionamiento del clúster.
- `/var/log/slurmctld.log`- Este es el registro del demonio de control de Slurm. AWS ParallelCluster no toma decisiones de escalado. Más bien, solo intenta lanzar recursos para satisfacer los requisitos de Slurm. Es útil para problemas de escalado y asignación, problemas relacionados con el trabajo y cualquier problema de lanzamiento y finalización relacionado con el programador.
- `/var/log/parallelcluster/compute_console_output`- Este registro registra la salida de la consola de un subconjunto de ejemplos de nodos de cómputo estáticos que han terminado inesperadamente. Utilice este registro si los nodos de cómputo estáticos terminan y los registros de los nodos de cómputo no están disponibles en él CloudWatch. El `compute_console_output` log contenido que recibe es el mismo cuando utiliza la consola EC2 o AWS CLI cuando recupera la salida de la consola de la instancia.

Estos son los registros clave de los nodos de cómputo:

- Este es el registro init del archivo `.init`. Contiene todos los comandos que se ejecutaron al configurar una instancia. Se usa para solucionar problemas de inicialización.
- Este es el registro de `clustermon`. Se ejecuta en cada nodo de cómputo para monitorizarlo en el raro caso de que el `clustermon` daemon del nodo principal esté desconectado. Se usa para solucionar problemas de terminación inesperados.
- `/var/log/slurmd.log`- Este es el registro del daemon de cómputo de Slurm. Se usa para solucionar problemas de inicialización y errores de computación.

Visualización de un error de **InsufficientInstanceCapacity** en **slurm_resume.log** cuando no puedo ejecutar un trabajo o en **clustermon.log** cuando no puedo crear un clúster

Si el clúster usa un Slurm programador, se está produciendo un problema de capacidad insuficiente. Si no hay suficientes instancias disponibles cuando se realiza la solicitud, se devuelve un error `InsufficientInstanceCapacity`.

Para la capacidad de las instancias estáticas, puede encontrar el error en el registro de `clustermon` en `/var/log/parallelcluster/clustermon`.

En cuanto a la capacidad dinámica de las instancias, puede encontrar el error en el registro de `ResumeProgram` en `/var/log/parallelcluster/slurm_resume.log`.

El ARN de tema tiene un aspecto similar al del ejemplo siguiente:

```
An error occurred (InsufficientInstanceCapacity) when calling the RunInstances/
CreateFleet operation...
```

Según su caso de uso, considere la posibilidad de utilizar uno de los siguientes métodos para evitar recibir este tipo de mensajes de error:

- Deshabilite el grupo de ubicación si está activado. Para obtener más información, consulte [Problemas con los grupos de ubicación y el lanzamiento de instancias](#).
- Reserve capacidad para las instancias y ejecútelas con las ODCR (reservas de capacidad bajo demanda). Para obtener más información, consulte [Inicio de instancias con ODCR \(reservas de capacidad bajo demanda\)](#).

- Configure varios recursos informáticos con distintos tipos de instancias. Si su carga de trabajo no requiere un tipo de instancia específico, puede aprovechar la conmutación por error rápida y con capacidad insuficiente con varios recursos informáticos. Para obtener más información, consulte [Conmutación por error rápida de capacidad insuficiente en el clúster de Slurm](#).
- Configura varios tipos de instancias en el mismo recurso informático y aprovecha la asignación de varios tipos de instancias. Para obtener más información sobre la configuración de varias instancias, consulte [Asignación de varios tipos de instancias con Slurm](#) y [Scheduling/SlurmQueues/ComputeResources/Instances](#).
- Mueva la cola a una zona de disponibilidad diferente cambiando el ID de subred en la configuración del clúster [Scheduling//SlurmQueues/Networking](#). [SubnetIds](#)
- Si su carga de trabajo no está estrechamente vinculada, distribuya la cola entre distintas zonas de disponibilidad. Para obtener más información acerca de la configuración de las subredes, consulte .

Solución de problemas de inicialización de nodos

En esta sección, se explica cómo solucionar los problemas de inicialización de los nodos. Esto incluye los problemas en los que el nodo no puede iniciar, encender o unirse a un clúster.

Temas

- [Nodo principal](#)
- [Nodos de computación](#)

Nodo principal

Registros aplicables:

- `/var/log/cfn-init.log`
- `/var/log/chef-client.log`
- `/var/log/parallelcluster/clustermgtd`
- `/var/log/parallelcluster/slurm_resume.log`
- `/var/log/slurmctld.log`

Compruebe los `/var/log/chef-client.log` registros `/var/log/cfn-init.log` y o los flujos de registro correspondientes. Estos registros contienen todas las acciones que se ejecutaron cuando

se configuró el nodo principal. La mayoría de los errores que se producen durante la configuración deberían incluir mensajes de error en el `/var/log/chef-client.log` registro. Si `OnNodeStart` se especifican `OnNodeConfigured` scripts en la configuración del clúster, compruebe que el script se ejecute correctamente a través de los mensajes de registro.

Cuando se crea un clúster, el nodo principal debe esperar a que los nodos de procesamiento se unan al clúster antes de poder unirse al clúster. Por este motivo, si los nodos informáticos no se unen al clúster, el nodo principal también falla. Puede seguir uno de estos procedimientos, dependiendo del tipo de notas de computación que utilice para solucionar problemas de este tipo:

Nodos de computación

- Registros aplicables:
 - `/var/log/cloud-init-output.log`
 - `/var/log/slurmd.log`
- Si se lanza un nodo de cómputo/`/var/log/cloud-init-output.log`, compruebe primero que contenga los registros de configuración similares al `/var/log/chef-client.log` registro del nodo principal. La mayoría de los errores que se producen durante la configuración deberían tener mensajes de error ubicados en el registro de `/var/log/cloud-init-output.log`. Si se especifican scripts previos o posteriores a la instalación en la configuración del clúster, compruebe que se hayan ejecutado correctamente.
- Si utiliza una AMI personalizada con modificaciones en la Slurm configuración, es posible que se produzca un error Slurm relacionado que impida que el nodo de procesamiento se una al clúster. Para ver los errores relacionados con el programador, consulte el registro. `/var/log/slurmd.log`

Nodos de computación dinámicos:

- Busque en el registro de `ResumeProgram (/var/log/parallelcluster/slurm_resume.log)` el nombre de su nodo de computación para ver si alguna vez se llamó a `ResumeProgram` con el nodo. (si nunca se ha llamado a `ResumeProgram`, puede consultar el registro de `slurmctld (/var/log/slurmctld.log)` para determinar si Slurm alguna vez intentó llamar a `ResumeProgram` con el nodo).
- Tenga en cuenta que los permisos incorrectos activados en `ResumeProgram` pueden provocar un error silencioso en `ResumeProgram`. Si utiliza una AMI personalizada con modificaciones en la configuración de `ResumeProgram`, compruebe que `ResumeProgram` sea propiedad del usuario de `slurm` y que tenga el permiso `744 (rwxr--r--)`.

- Si se llama a `ResumeProgram`, compruebe si se ha lanzado una instancia para el nodo. Si no se lanzó ninguna instancia, verá un mensaje de error que describe el error de lanzamiento.
- Si se lanza la instancia, es posible que haya habido un problema durante el proceso de configuración. Debería ver la dirección IP privada y el ID de instancia correspondientes en el registro de `ResumeProgram`. Además, puede consultar los registros de configuración correspondientes a la instancia específica. Para obtener más información acerca de la solución de problemas relacionados con un nodo de computación, consulte la siguiente sección.

Nodos de computación estáticos:

- Compruebe el registro `clustermgtd (/var/log/parallelcluster/clustermgtd)` para ver si se han lanzado instancias para el nodo. Si no se lanzaron, debería haber un mensaje de error claro que detalle el error de lanzamiento.
- Si se lanza la instancia, hay algún problema durante el proceso de configuración. Debería ver la dirección IP privada y el ID de instancia correspondientes en el registro de `ResumeProgram`. Además, puede consultar los registros de configuración correspondientes a la instancia específica.

Nodos de computación respaldados por instancias de spot:

- Si es la primera vez que utiliza instancias de spot y el trabajo permanece en estado de PD (pendiente), compruebe el `/var/log/parallelcluster/slurm_resume.log` archivo. Probablemente encuentre un error como el siguiente:

```
2022-05-20 13:06:24,796 - [slurm_plugin.common:add_instances_for_nodes] - ERROR -  
Encountered exception when launching instances for nodes (x1) ['spot-dy-t2micro-2']:  
An error occurred (AuthFailure.ServiceLinkedRoleCreationNotPermitted) when calling  
the RunInstances operation: The provided credentials do not have permission to  
create the service-linked role for EC2 Spot Instances.
```

Al utilizar instancias de spot, en la cuenta se debe incluir un rol vinculado a un servicio `AWSServiceRoleForEC2Spot`. Para crear este rol en su cuenta mediante el AWS CLI, ejecute el siguiente comando:

```
$ aws iam create-service-linked-role --aws-service-name spot.amazonaws.com
```

Para obtener más información, consulte [Uso de instancias de spot](#) la Guía del AWS ParallelCluster usuario y la [función vinculada al servicio para las solicitudes de instancias puntuales](#) de la Guía del usuario de Amazon EC2.

Solución de problemas de sustituciones y terminaciones inesperadas de nodos

En esta sección se continúa analizando cómo puede solucionar problemas relacionados con los nodos, específicamente cuando un nodo se reemplaza o se cierra inesperadamente.

- Registros aplicables:
 - `/var/log/parallelcluster/clustermgtd` (nodo principal)
 - `/var/log/slurmctld.log` (nodo principal)
 - `/var/log/parallelcluster/computemgtd` (nodo de computación)

Nodos sustituidos o finalizados de forma inesperada:

- Para comprobar si `clustermgtd` ha sustituido o finalizado un nodo, compruebe el registro . Tenga en cuenta que `clustermgtd` gestiona todas las acciones normales de mantenimiento del nodo.
- Si `clustermgtd` se reemplaza o se cierra el nodo, debería haber un mensaje que detalle por qué se realizó esta acción en el nodo. Si el motivo está relacionado con el programador (por ejemplo, el nodo está en DOWN), consulte el registro `slurmctld` para obtener más detalles. Si el motivo está relacionado con Amazon EC2, debería haber un mensaje informativo en el que se detalle el problema relacionado con Amazon EC2 que requirió la sustitución.
- Si `clustermgtd` no terminó el nodo, compruebe primero si se trataba de una terminación prevista por parte de Amazon EC2, más específicamente una terminación puntual. `computemgtd`, que se ejecuta en un nodo de cómputo, también puede cerrar un nodo si `clustermgtd` se determina que está en mal estado. Compruebe el registro de `computemgtd` (`/var/log/parallelcluster/computemgtd`) para ver si `computemgtd` ha terminado el nodo.

Los nodos fallaron

- Compruebe el registro de `slurmctld` (`/var/log/slurmctld.log`) para ver por qué ha fallado un trabajo o un nodo. Tenga en cuenta que los trabajos se vuelven a poner en cola automáticamente si se produce un error en un nodo.
- Si `slurm_resume` informa de que el nodo se ha lanzado y, después de varios minutos, `clustermgtd` informa de que no hay ninguna instancia correspondiente en Amazon EC2 para ese nodo, es posible que el nodo produzca un error durante la configuración. Para recuperar el registro de un compute (`/var/log/cloud-init-output.log`), siga estos pasos:
 - Envía un trabajo para permitir que Slurm active un nuevo nodo.
 - Una vez que se inicie el nodo, habilite la protección de terminación mediante este comando.

```
$ aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --disable-api-termination
```

- Recupere la salida de la consola del nodo con este comando.

```
$ aws ec2 get-console-output --instance-id i-1234567890abcdef0 --output text
```

Reemplazar, terminar o apagar instancias y nodos problemáticos

- Registros aplicables:
 - `/var/log/parallelcluster/clustermgtd` (nodo principal)
 - `/var/log/parallelcluster/slurm_suspend.log` (nodo principal)
- En la mayoría, `clustermgtd` administra todas las acciones de finalización de instancias esperadas. Consulte el registro `clustermgtd` para ver por qué no se ha podido sustituir o finalizar un nodo.
- En el caso de los nodos dinámicos que no superan el [Propiedades de SlurmSettings](#), consulte el registro de `SuspendProgram` para ver si lo llamó `slurmctld` con el nodo específico como argumento. Tenga en cuenta que `SuspendProgram` no realiza ninguna acción específica. Más bien, solo se encarga de registrar cuando se le llama. La terminación de todas las instancias y el restablecimiento de `NodeAddr` los realiza `clustermgtd`. Slurm vuelve a poner los nodos en un `POWER_SAVING` estado posterior `SuspendTimeout` automáticamente.
- Si los nodos de cómputo fallan continuamente debido a errores de arranque, compruebe si se están iniciando con la tecla `Enabled`. [Modo protegido de clúster Slurm](#) Si el modo protegido no está activado, modifique la configuración del modo protegido para activarlo. Solucione los problemas y corrija el script de arranque.

Estado de la cola (partición) **Inactive**

Si ejecuta `sinfo` y el resultado muestra colas con el `AVAIL` estado de `inact`, es posible que su clúster [Modo protegido de clúster Slurm](#) esté habilitado y que la cola se haya establecido en ese `INACTIVE` estado durante un período de tiempo predefinido.

Solución de otros problemas de nodos y trabajos conocidos

Otro tipo de problema conocido es que es AWS ParallelCluster posible que no se puedan asignar los trabajos ni tomar decisiones de escalado. En este tipo de problemas, AWS ParallelCluster solo inicia, finaliza o mantiene los recursos de acuerdo con las instrucciones de Slurm. En el caso de estos problemas, consulte el `slurmctld` registro para solucionarlos.

Problemas con los grupos de ubicación y el lanzamiento de instancias

Para obtener la latencia entre nodos más baja, use un grupo de ubicación. Un grupo de ubicación garantiza que las instancias estén en la misma red troncal. Si no hay suficientes instancias disponibles cuando se realiza la solicitud, se devuelve un error `InsufficientInstanceCapacity`. Para reducir la posibilidad de recibir un error [SlurmQueues](#) al utilizar grupos con ubicación en clúster, establezca el parámetro [Networking](#) en [PlacementGroup](#) y establezca el parámetro [Enabled](#) en `false`.

Para tener un control adicional sobre la capacidad de acceso, considere [lanzar instancias con ODCR \(reservas de capacidad bajo demanda\)](#).

Para obtener más información, consulte [Solución de problemas de lanzamiento de instancias y Roles y limitaciones de los grupos de ubicación](#) en la Guía de usuario para instancias Linux de Amazon EC2.

Directorios que no se pueden reemplazar

Los siguientes directorios se comparten entre los nodos y no se pueden reemplazar.

- `/home`- Esto incluye la carpeta de inicio del usuario predeterminada (`/home/ec2_user` en Amazon Linux RedHat, `/home/centos` en Centos y `/home/ubuntu` en Ubuntu).
- Esto incluye Intel MPI, Intel Parallel Studio y archivos relacionados.

- `/opt/slurm` - Esto incluye Slurm Workload Manager y los archivos relacionados. (Condicional, solo si `Scheduler: slurm`).

Solución de problemas de NICE DCV

Temas

- [Registros para NICE DCV](#)
- [Problemas con Ubuntu NICE DCV](#)

Registros para NICE DCV

Los registros de NICE DCV se escriben en los archivos del directorio `/var/log/dcv/`. Revisar estos registros puede ayudar a solucionar problemas.

El tipo de instancia debe tener al menos 1,7 gibibytes (GiB) de RAM para ejecutar NICE DCV. Los tipos de instancias nano y micro no tienen suficiente memoria para ejecutar NICE DCV.

AWS ParallelCluster crea flujos de registro NICE DCV en grupos de registros. Puede ver estos registros en los paneles personalizados de la CloudWatch consola o en los grupos de registros. Para obtener más información, consulte [Integración con Amazon CloudWatch Logs](#) y [Panel de Amazon CloudWatch](#).

Problemas con Ubuntu NICE DCV

Al ejecutar GNOME Terminal en una sesión NICE DCV en Ubuntu, es posible que no tenga acceso automáticamente al entorno de usuario AWS ParallelCluster que está disponible a través del intérprete de comandos de inicio de sesión. El entorno de usuario proporciona módulos de entorno, como `openmpi` o `intelmpi`, y otros ajustes de usuario.

La configuración predeterminada de la Terminal de GNOME impide que el intérprete de comandos se inicie como un intérprete de comandos de inicio de sesión. Esto significa que los perfiles de shell no se obtienen automáticamente y el entorno AWS ParallelCluster de usuario no se carga.

Para obtener correctamente el perfil de shell y acceder al entorno AWS ParallelCluster de usuario, realice una de las siguientes acciones:

- Cambie la configuración predeterminada del terminal:
 1. Seleccione el menú Editar en la terminal de GNOME.

2. Seleccione Preferencias y, a continuación, Perfiles.
 3. Elija Comando y seleccione Ejecutar comando como intérprete de comandos de inicio de sesión.
 4. Abrir una nueva terminal.
- Utilice la línea de comandos para obtener los perfiles disponibles:

```
$ source /etc/profile && source $HOME/.bashrc
```

Solución de problemas en clústeres con AWS Batch integración

Esta sección es relevante para los clústeres con integración de AWS Batch planificadores.

Temas

- [Problemas con el nodo principal](#)
- [Problemas informáticos](#)
- [Errores en los trabajos](#)
- [Error de tiempo de espera de conexión en la URL del punto de conexión](#)

Problemas con el nodo principal

Puede solucionar los problemas de configuración del nodo principal de la misma manera que con un Slurm clúster (excepto en el caso de registros Slurm específicos). Para obtener más información sobre estos problemas, consulte [Nodo principal](#).

Problemas informáticos

AWS Batch gestiona los aspectos de escalado y computación de sus servicios. Si tiene problemas relacionados con la informática, consulte la documentación AWS Batch [de solución de problemas](#) para obtener ayuda.

Errores en los trabajos

Si se produce un error en un trabajo, puede ejecutar el comando [awsbcout](#) para recuperar el resultado del trabajo. También puedes ejecutar el [awsbstat](#) comando para obtener un enlace a los registros de trabajos almacenados por Amazon CloudWatch.

Error de tiempo de espera de conexión en la URL del punto de conexión

Si los trabajos paralelos de varios nodos fallan y se produce un error: `Connect timeout on endpoint URL`

- En el registro `awsbout` de salida, compruebe que el trabajo sea de varios nodos en paralelo a la salida: `Detected 3/3 compute nodes. Waiting for all compute nodes to start.`
- Compruebe si la subred de los nodos de procesamiento es pública.

Los trabajos paralelos de varios nodos no admiten el uso de subredes públicas cuando se utilizan AWS Batch in. AWS ParallelCluster Use una subred privada para sus tareas y nodos de cómputo. Para obtener más información, consulte [Entornos informáticos](#) en la Guía del usuario de AWS Batch . Para configurar una subred privada para los nodos de procesamiento, consulte. [AWS ParallelCluster con el programador AWS Batch](#)

Solución de problemas de integración multiusuario con Active Directory

Esta sección es relevante para los clústeres integrados en un Active Directory.

Si la característica de integración de Active Directory no funciona como se esperaba, los registros SSSD pueden proporcionar información de diagnóstico útil. Estos registros se encuentran en el directorio `/var/log/sss` de los nodos del clúster. De forma predeterminada, también se almacenan en el grupo de CloudWatch registros de Amazon de un clúster.

Temas

- [Solución de problemas específicos de Active Directory](#)
- [Cómo habilitar el modo de depuración](#)
- [Cómo pasar de LDAPS a LDAP](#)
- [Cómo deshabilitar la verificación del certificado del servidor LDAPS](#)
- [¿Cómo iniciar sesión con una clave SSH en lugar de una contraseña?](#)
- [Cómo restablecer una contraseña de usuario y contraseñas caducadas](#)
- [¿Cómo comprobar el dominio al que se ha unido](#)
- [Cómo solucionar los problemas con .](#)
- [Cómo comprobar que la integración con Active Directory funciona](#)

- [Cómo solucionar problemas al iniciar sesión en los nodos de cómputo](#)
- [Problemas conocidos con los trabajos de SimCenter StarCCM+ en un entorno multiusuario](#)
- [Problemas conocidos relacionados con la resolución del nombre de usuario](#)
- [Cómo resolver los problemas de creación del directorio principal](#)

Solución de problemas específicos de Active Directory

Esta sección es relevante para la solución de problemas específicos de un tipo de Active Directory.

AD sencillo

- El valor de `DomainReadOnlyUser` debe coincidir con la búsqueda base del directorio Simple AD para los usuarios:

```
cn=ReadOnlyUser,cn=Users,dc=corp,dc=example,dc=com
```

Nota `cn` para `Users`.

- El usuario administrador predeterminado es `Administrator`.
- `Ldapsearch` requiere el nombre de NetBIOS antes del nombre de usuario.

La sintaxis / debe ser la siguiente:

```
$ ldapsearch -x -D "corp\\Administrator" -w "Password" -H ldap://192.0.2.103 \
-b "cn=Users,dc=corp,dc=example,dc=com"
```

AWS Managed Microsoft AD

- El `DomainReadOnlyUser` valor debe coincidir con la búsqueda base del AWS Managed Microsoft AD directorio para los usuarios:

```
cn=ReadOnlyUser,ou=Users,ou=CORP,dc=corp,dc=example,dc=com
```

- El usuario administrador predeterminado es `Admin`.
- La sintaxis / debe ser la siguiente:

```
$ ldapsearch -x -D "Admin" -w "Password" -H ldap://192.0.2.103 \
-b "ou=Users,ou=CORP,dc=corp,dc=example,dc=com"
```

Cómo habilitar el modo de depuración

Los registros de depuración de SSSD pueden ser útiles para solucionar problemas. Para habilitar el modo de depuración, debe actualizar el clúster con los siguientes cambios realizados en la configuración del clúster:

```
DirectoryService:
  AdditionalSssdConfigs:
    debug_level: "0x1ff"
```

Cómo pasar de LDAPS a LDAP

No se recomienda pasar del LDAPS (LDAP con TLS/SSL) al LDAP porque el LDAP por sí solo no proporciona ningún tipo de cifrado. Sin embargo, puede resultar útil para realizar pruebas y solucionar problemas.

Puede restaurar el clúster a su configuración anterior actualizándolo con la definición de configuración anterior.

Para pasar de LDAPS a LDAP, debe actualizar el clúster con los siguientes cambios en la configuración del clúster:

```
DirectoryService:
  LdapTlsReqCert: never
  AdditionalSssdConfigs:
    ldap_auth_disable_tls_never_use_in_production: True
```

Cómo deshabilitar la verificación del certificado del servidor LDAPS

Puede resultar útil deshabilitar temporalmente la verificación del certificado del servidor LDAPS en el nodo principal para realizar pruebas o solucionar problemas.

Puede restaurar el clúster a su configuración anterior actualizándolo con la definición de configuración anterior.

Para deshabilitar la verificación del certificado del servidor LDAPS, debe actualizar el clúster con los siguientes cambios en la configuración del clúster:

```
DirectoryService:
```

```
LdapTlsReqCert: never
```

¿Cómo iniciar sesión con una clave SSH en lugar de una contraseña?

La clave SSH se crea `/home/$user/.ssh/id_rsa` después de iniciar sesión por primera vez con una contraseña. Para iniciar sesión con la clave SSH, debe iniciar sesión con su contraseña, copiar la clave SSH localmente y, a continuación, utilizarla para realizar SSH sin contraseña, como de costumbre:

```
$ ssh -i $LOCAL_PATH_TO_SSH_KEY $username@$head_node_ip
```

Cómo restablecer una contraseña de usuario y contraseñas caducadas

Si un usuario pierde el acceso a un clúster, es [posible que su AWS Managed Microsoft AD contraseña haya caducado](#).

Para restablecer la contraseña, ejecute el siguiente comando con un usuario y un rol que tengan permiso de escritura en el directorio:

```
$ aws ds reset-user-password \  
  --directory-id "d-abcdef01234567890" \  
  --user-name "USER_NAME" \  
  --new-password "NEW_PASSWORD" \  
  --region "region-id"
```

Si restablece la contraseña de [DirectoryService/DomainReadOnlyUser](#):

1. Asegúrese de actualizar el [PasswordSecretArn](#) secreto [DirectoryService](#)/con la nueva contraseña.
2. Actualice el clúster para el nuevo valor secreto:
 - a. Detenga la flota de computación con el comando `pcluster update-compute-fleet`.
 - b. A continuación, ejecute el siguiente comando desde el nodo principal del clúster.

```
$ sudo /opt/parallelcluster/scripts/directory_service/  
update_directory_service_password.sh
```

Tras restablecer la contraseña y actualizar el clúster, se debe restablecer el acceso del usuario al clúster.

Para obtener más información, consulte [Crear un usuario](#) en la Guía de administración de AWS Directory Service .

¿Cómo comprobar el dominio al que se ha unido

El siguiente comando debe ejecutarse desde una instancia que esté unida al dominio, no desde el nodo principal.

```
$ realm list corp.example.com \  
type: kerberos \  
realm-name: CORP.EXAMPLE.COM \  
domain-name: corp.example.com \  
configured: kerberos-member \  
server-software: active-directory \  
client-software: sssd \  
required-package: oddjob \  
required-package: oddjob-mkhomedir \  
required-package: sssd \  
required-package: adcli \  
required-package: samba-common-tools \  
login-formats: %U \  
login-policy: allow-realm-logins
```

Cómo solucionar los problemas con .

Cuando la comunicación LDAPS no funciona, puede deberse a errores en la comunicación TLS, que a su vez pueden deberse a problemas con los certificados.

Notas acerca de los certificados:

- El certificado especificado en la configuración del clúster `LdapTlsCaCert` debe ser un paquete de certificados PEM que contenga los certificados de toda la cadena de certificados de autoridad (CA) que emitió los certificados para los controladores de dominio.
- Un paquete de certificados PEM es un archivo compuesto por la concatenación de certificados PEM.
- Un certificado en formato PEM (normalmente utilizado en Linux) equivale a un certificado en formato DER base64 (normalmente exportado por Windows).
- Si el certificado para los controladores de dominio lo emite una CA subordinada, el paquete de certificados debe contener el certificado de la CA subordinada y raíz.

Pasos de verificación para solucionar problemas:

En los siguientes pasos de verificación se supone que los comandos se ejecutan desde el nodo principal del clúster y que se puede acceder al controlador de dominio en **SERVER:PORT** él.

Para solucionar un problema relacionado con los certificados, siga estos pasos de verificación:

Pasos de verificación:

1. Compruebe la conexión a los controladores de dominio de Active Directory:

Comprobar que puede conectarse a un controlador de dominio. Si este paso se realiza correctamente, la conexión SSL al controlador de dominio se realiza correctamente y se verifica el certificado. El problema no está relacionado con los certificados.

Si este paso no funciona, continúa con la siguiente verificación.

```
$ openssl s_client -connect SERVER:PORT -CAfile PATH_TO_CA_BUNDLE_CERTIFICATE
```

2. Compruebe la verificación del certificado:

Compruebe que el paquete de certificados de CA local pueda validar el certificado proporcionado por el controlador de dominio. Si este paso se realiza correctamente, el problema no está relacionado con los certificados, sino con otros problemas de red.

Si este paso no funciona, continúa con la siguiente verificación.

```
$ openssl verify -verbose -  
CAfile PATH_TO_CA_BUNDLE_CERTIFICATE PATH_TO_A_SERVER_CERTIFICATE
```

3. Compruebe el certificado proporcionado por los controladores de dominio de Active Directory:

Compruebe que el contenido del certificado proporcionado por los controladores de dominio es el esperado. Si este paso se realiza correctamente, es probable que tenga problemas con el certificado de CA utilizado para verificar los controladores. Continúe con el siguiente paso de solución de problemas.

Si este paso no funciona, debe corregir el certificado emitido para los controladores de dominio y volver a ejecutar los pasos de solución de problemas.

```
$ openssl s_client -connect SERVER:PORT -showcerts
```

4. Compruebe el contenido de un certificado:

Compruebe que el contenido del certificado que proporcionan los controladores de dominio es el esperado. Si este paso se realiza correctamente, es probable que tenga problemas con el certificado de CA utilizado para verificar los controladores. Continúe con el siguiente paso de solución de problemas.

Si este paso no funciona, debe corregir el certificado emitido para los controladores de dominio y volver a ejecutar los pasos de solución de problemas.

```
$ openssl s_client -connect SERVER:PORT -showcerts
```

5. Compruebe el contenido del paquete de certificados de CA local:

Compruebe que el contenido del paquete de certificados de CA local utilizado para validar el certificado del controlador de dominio es el esperado. Si este paso se realiza correctamente, es probable que tenga problemas con el certificado proporcionado por los controladores de dominio.

Si este paso no funciona, debe corregir el paquete de certificados de CA emitido para los controladores de dominio y volver a ejecutar los pasos de solución de problemas.

```
$ openssl x509 -in PATH_TO_A_CERTIFICATE -text
```

Cómo comprobar que la integración con Active Directory funciona

Si las dos comprobaciones siguientes se realizan correctamente, la integración con Active Directory funciona.

verificaciones

1. Puede detectar los usuarios definidos en el directorio:

Desde el nodo principal del clúster, como `ec2-user`:

```
$ getent passwd $ANY_AD_USER
```

2. Puede acceder mediante SSH al nodo principal proporcionando la contraseña de usuario:

```
$ ssh $ANY_AD_USER@$HEAD_NODE_IP
```

Si la primera comprobación falla, esperamos que la segunda también falle.

Solución de problemas adicionales

- Compruebe que el usuario existe en el directorio.
- Habilitación del registro de depuración de
- Considere la posibilidad de deshabilitar temporalmente el cifrado [pasando de LDAPS a LDAP para descartar problemas con el LDAPS](#).

Cómo solucionar problemas al iniciar sesión en los nodos de cómputo

Esta sección es relevante para iniciar sesión en los nodos de cómputo de los clústeres integrados con Active Directory.

Con AWS ParallelCluster, los inicios de sesión con contraseña en los nodos de cómputo del clúster están deshabilitados por diseño.

Todos los usuarios deben usar su propia clave SSH para iniciar sesión en los nodos de procesamiento.

Los usuarios pueden recuperar su clave SSH en el nodo principal tras la primera autenticación (por ejemplo, al iniciar sesión), si [GenerateSshKeysForUsers](#) está habilitada en la configuración del clúster.

Cuando los usuarios se autentican en el nodo principal por primera vez, pueden recuperar las claves SSH que se generan automáticamente para ellos como usuarios del directorio. También se crean los directorios principales para el usuario. Esto también puede ocurrir la primera vez que un sudo-usuario cambia a un usuario del nodo principal.

Si un usuario no ha iniciado sesión en el nodo principal, las claves SSH no se generan y el usuario no podrá iniciar sesión en los nodos de cálculo.

Problemas conocidos con los trabajos de SimCenter StarCCM+ en un entorno multiusuario

Esta sección es relevante para los trabajos lanzados en un entorno multiusuario por el software de dinámica de fluidos computacional Simcenter StarCCM+ de Siemens.

Si ejecuta tareas de StarCCM+ v16 configuradas para utilizar el IntelMPI integrado, los procesos del MPI se iniciarán de forma predeterminada mediante SSH.

Debido a un error conocido de [Slurm que hace que la resolución del nombre de usuario sea incorrecta, es posible que los trabajos fallen y se produzca un error](#) como el siguiente. `error setting up the bootstrap proxies` Este error solo afecta a las AWS ParallelCluster versiones 3.1.1 y 3.1.2.

Para evitar que esto ocurra, obligue a IntelMPI a usar Slurm como método de arranque de MPI. Exporte la variable de entorno `I_MPI_HYDRA_BOOTSTRAP=slurm` al script de trabajo que lanza StarCCM+, tal y como se describe en la [documentación oficial de IntelMPI](#).

Problemas conocidos relacionados con la resolución del nombre de usuario

Esta sección es relevante para recuperar los nombres de usuario en los trabajos.

Debido a un [error conocido en Slurm](#), el nombre de usuario que se recupera en un proceso de trabajo puede ser el de ejecutar un trabajo nobody sin él. `srun` Este error solo afecta a AWS ParallelCluster las versiones 3.1.1 y 3.1.2.

Por ejemplo, si ejecuta el comando `sbatch --wrap 'srun id'` como usuario del directorio, se devuelve el nombre de usuario correcto. Sin embargo, si lo ejecuta `sbatch --wrap 'id'` como usuario del directorio, nobody es posible que se devuelva como nombre de usuario.

Puede utilizar las siguientes soluciones alternativas.

1. Inicie su trabajo con, ' `srun` ' en lugar de ' `sbatch` ', si es posible.
2. Habilite la enumeración SSSD configurando las configuraciones en la [AdditionalSssdconfiguración](#) del clúster de la siguiente manera.

```
AdditionalSssdConfigs:  
  enumerate: true
```

Cómo resolver los problemas de creación del directorio principal

Esta sección es relevante para los problemas de creación del directorio principal.

Si ve errores como el que se muestra en el siguiente ejemplo, significa que no se creó un directorio principal para usted cuando inició sesión por primera vez en el nodo principal. O bien, no se creó un directorio principal para usted cuando cambió por primera vez de usuario de sudoer a usuario de Active Directory en el nodo principal.

```
$ ssh AD_USER@$HEAD_NODE_IP
/opt/parallelcluster/scripts/generate_ssh_key.sh failed: exit code 1

  _|  _|_ )
 _| (    /  Amazon Linux 2 AMI
  _|\__|__|

https://aws.amazon.com/amazon-linux-2/
Could not chdir to home directory /home/PclusterUser85: No such file or directory
```

El error al crear el directorio principal puede deberse a los oddjob-mkhomedir paquetes oddjob y instalados en el nodo principal del clúster.

Sin un directorio principal y una clave SSH, el usuario no puede enviar trabajos o SSH a los nodos del clúster.

Si necesita los oddjob paquetes en su sistema, compruebe que el oddjobd servicio se esté ejecutando y actualice los archivos de configuración de PAM para asegurarse de que se ha creado el directorio principal. Para ello, ejecute los comandos en el nodo principal como se muestra en el siguiente ejemplo.

```
sudo systemctl start oddjobd
sudo authconfig --enablemkhomedir --updateall
```

Si no necesita los oddjob paquetes en su sistema, desinstálelos y actualice los archivos de configuración de PAM para asegurarse de que se ha creado el directorio principal. Para ello, ejecute los comandos en el nodo principal como se muestra en el siguiente ejemplo.

```
sudo yum remove -y oddjob oddjob-mkhomedir
sudo authconfig --enablemkhomedir --updateall
```

Solución de problemas con las AMI de

Al utilizar una AMI personalizada, puede ver las siguientes advertencias:

```
"validationMessages": [  
  {  
    "level": "WARNING",  
    "type": "CustomAmiTagValidator",  
    "message": "The custom AMI may not have been created by pcluster. You can ignore  
this warning if the AMI is shared or copied from another pcluster AMI. If the  
AMI is indeed not created by pcluster, cluster creation will fail. If the cluster  
creation fails, please go to https://docs.aws.amazon.com/parallelcluster/latest/ug/  
troubleshooting.html#troubleshooting-stack-creation-failures for troubleshooting."  
  },  
  {  
    "level": "WARNING",  
    "type": "AmiOsCompatibleValidator",  
    "message": "Could not check node AMI ami-0000012345 OS and cluster OS alinux2  
compatibility, please make sure they are compatible before cluster creation and update  
operations."  
  }  
]
```

Si está seguro de que se está utilizando la AMI correcta, puede ignorar estas advertencias.

Si no quiere ver estas advertencias en el futuro, etiquete la AMI personalizada con las siguientes etiquetas, donde *my-os* es una de las versiones de alinux2, ubuntu2204, ubuntu2004, centos7 o rhel8 y *"3.7.0"* es la versión de pcluster en uso:

```
$ aws ec2 create-tags \  
  --resources ami-yourcustomAmi \  
  --tags Key="parallelcluster:version",Value="3.7.0"  
  Key="parallelcluster:os",Value="my-os"
```

Solución de problemas cuando se agota el tiempo de espera de una actualización del clúster cuando no se está ejecutando **cfn-hup**

El script auxiliar `cfn-hup` es un daemon que detecta cambios en los metadatos de recursos y ejecuta acciones especificadas por el usuario cuando se detecta un cambio. Esto le permite llevar a cabo actualizaciones de la configuración en las instancias de Amazon EC2 que se están ejecutando a través de la acción de la API `UpdateStack`.

Actualmente, el `cfn-hup` daemon lo lanza el `supervisord`. Pero después del lanzamiento, el `cfn-hup` proceso se separa del `supervisord` control. Si un actor externo acaba con el daemon de `cfn-hup`, no se reinicia automáticamente. Si `cfn-hup` no se está ejecutando, durante una actualización del clúster, la CloudFormation pila inicia el proceso de actualización según lo previsto, pero el procedimiento de actualización no se activa en el nodo principal y, finalmente, se agota el tiempo de espera de la pila. En los registros del clúster `/var/log/chef-client`, puede ver que la receta de actualización nunca se invoca.

Compruébelo y reinícielo **cfn-hup** en caso de errores

1. En el nodo principal, compruebe si `cfn-hup` se está ejecutando:

```
$ ps aux | grep cfn-hup
```

2. Compruebe el `cfn-hup` registro `/var/log/cfn-hup.log` y `/var/log/supervisord.log` el nodo principal.
3. Si `cfn-hup` no se está ejecutando, intenta reiniciarlo ejecutando:

```
$ sudo /opt/parallelcluster/pyenv/versions/cookbook_virtualenv/bin/supervisorctl start cfn-hup
```

Solución de problemas de redes en bastidor

Problemas con los clústeres en una única subred pública

Compruébelo `cloud-init-output.log` desde uno de los nodos de cómputo. Si encuentra algo como lo siguiente que indica que el nodo está atascado en la inicialización de Slurm, lo más probable es que se deba a la falta de un punto de conexión de VPC de DynamoDB. Añada el punto

de conexión de DynamoDB. Para más información, consulte [AWS ParallelCluster en una subred individual sin acceso a Internet](#).

```
ruby_block[retrieve compute node info] action run[2022-03-11T17:47:11+00:00] INFO:
  Processing ruby_block[retrieve compute node info] action run (aws-parallelcluster-
slurm::init line 31)
```

No se pudo actualizar el clúster al realizar una acción **onNodeUpdated** personalizada

Cuando se produce un error en un [OnNodeUpdated](#)script [HeadNodeCustomActions](#)//, se produce un error en la actualización y el script no se ejecuta en el momento de la reversión. Es su responsabilidad realizar manualmente las limpiezas necesarias una vez finalizada la reversión. Por ejemplo, si el OnNodeUpdated script cambia el estado de un campo en un archivo de configuración (por ejemplo, de true a false) y, a continuación, se produce un error, tendrá que restaurar manualmente el valor de ese campo al estado anterior a la actualización (por ejemplo, a false true). Para obtener más información, consulte [Acciones de arranque personalizadas](#).

Visualización de errores en la configuración personalizada de Slurm

A partir de AWS ParallelCluster la versión 3.6.0, ya no puede centrarse en scripts individuales prolog o epilog scripts incluyéndolos en una configuración personalizadaSlurm. En AWS ParallelCluster la versión 3.6.0 y versiones posteriores, debe buscar los scripts personalizados prolog y los epilog scripts en las carpetas Prolog y Epilog correspondientes. Estas carpetas están configuradas de forma predeterminada para que apunten a:

- Prolog apunta a /opt/slurm/etc/scripts/prolog.d/.
- Epilog apunta a /opt/slurm/etc/scripts/epilog.d/.

Le recomendamos que mantenga el guion del 90_plcluster_health_check_manager prólogo y el guion del 90_pcluster_noop epílogo en su lugar.

Slurm ejecuta los scripts en orden alfabético inverso. Tanto la carpeta Prolog como la Epilog deben contener un archivo como mínimo. Para obtener más información, consulte [Slurm prolog y epilog](#) y [Personalización de la configuración de Slurm](#).

Alarmas de clúster

La supervisión del estado del clúster es esencial para garantizar un rendimiento óptimo. AWS ParallelCluster le permite monitorear múltiples alarmas CloudWatch basadas en el nodo principal del clúster.

En esta sección se proporcionan detalles sobre cada tipo de alarma de clúster del nodo principal, incluidas sus convenciones de nomenclatura, las condiciones específicas que activan las alarmas y las medidas sugeridas para la solución de problemas.

La convención de nomenclatura de las alarmas de clúster es `CLUSTER_NAME-COMPONENT-METRIC`, `mycluster-HeadNode-Cpu` p. ej.

- `CLUSTER_NAME-HeadNode`: indica el estado general del nodo principal. Se muestra en rojo si al menos una de las siguientes alarmas está activada.
- `CLUSTER_NAME-HeadNode-Health`: rojo si hay al menos un fallo en el EC2 Health Check. En caso de alarma, le sugerimos que consulte la sección [Solución de problemas en los que las comprobaciones de estado no se han realizado](#) correctamente.
- `CLUSTER_NAME-HeadNode-Cpu`: rojo si el uso de la CPU es superior al 90%. En caso de alarma, compruebe los procesos que más consumen la CPU `ps -aux --sort=-%cpu | head -n 10`.
- `CLUSTER_NAME-HeadNode-Mem`: rojo si la utilización de la memoria es superior al 90%. En caso de alarma, compruebe con atención los procesos que más consumen memoria `ps -aux --sort=-%mem | head -n 10`.
- `CLUSTER_NAME-HeadNode-Disk`: rojo si el espacio ocupado en disco es superior al 90% en la ruta `/`. En caso de alarma, compruebe las carpetas que ocupan la mayor parte del espacio. `du -h --max-depth=2 /dev/null | sort -hr`

Compatibilidad adicional

Para ver una lista de problemas conocidos, consulta la página principal de la [GitHub wiki](#) o la página de [problemas](#).

Para problemas más urgentes, contacta AWS Support o abre un [nuevo GitHub número](#).

AWS ParallelCluster política de apoyo

AWS ParallelCluster admite varias versiones al mismo tiempo. Cada AWS ParallelCluster lanzamiento tiene una fecha de fin de vida útil (EOSL) programada. A partir de la fecha de EOSL no se proporcionará más soporte ni mantenimiento para ese lanzamiento.

AWS ParallelCluster utiliza un esquema de `major.minor.patch` versiones. Las nuevas versiones secundarias de la última versión principal incluyen nuevas características, mejoras de rendimiento, actualizaciones de seguridad y correcciones de errores. Las versiones secundarias son compatibles con versiones anteriores de una versión principal. Para los problemas críticos, AWS proporciona correcciones mediante versiones de parches, pero solo para las versiones secundarias más recientes de las versiones que no han llegado a la EOSL. Si desea utilizar las actualizaciones de una versión nueva, debe actualizar a la nueva versión secundaria o a la nueva versión de parche.

AWS ParallelCluster versiones	Fecha de fin de compatibilidad (EOSL)
3.0. <i>x</i>	31/03/2023
3.1. <i>x</i>	31/8/2023
3.2. <i>x</i>	31/1/2024
3.3. <i>x</i>	31/05/2024
3.4. <i>x</i>	28/06/2024
3.5. <i>x</i>	31/8/2024
3.6. <i>x</i>	30/11/2024
3.7. <i>x</i>	28/02/2025
3.8. <i>x</i>	30/06/2025
3.9. <i>x</i>	09/05/2025

Seguridad en AWS ParallelCluster

La seguridad en la nube de AWS es la mayor prioridad. Como cliente de AWS, se beneficia de una arquitectura de red y un centro de datos que se han diseñado para satisfacer los requisitos de seguridad de las organizaciones más exigentes.

La seguridad es una responsabilidad compartida entre AWS y usted. El [modelo de responsabilidad compartida](#) describe esto como seguridad de la nube y seguridad en la nube:

- Seguridad de la nube: AWS es responsable de proteger la infraestructura que ejecuta los servicios de AWS en la nube de AWS. AWS también 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 conformidad de](#) . Para obtener información sobre los programas de conformidad que se aplican a AWS ParallelCluster, consulte [Servicios de AWS en el ámbito del programa de conformidad](#).
- Seguridad en la nube: su responsabilidad viene determinada por el servicio o servicios específicos de AWS que utilice. También es responsable de otros factores varios, incluida la confidencialidad de los datos, los requisitos de la empresa y la legislación y los reglamentos aplicables.

Esta documentación describe cómo aplicar el modelo de responsabilidad compartida cuando se utiliza AWS ParallelCluster. En los siguientes temas, se le mostrará cómo configurar AWS ParallelCluster para satisfacer sus objetivos de seguridad y conformidad. También aprenderá a utilizar AWS ParallelCluster de manera que le ayudará a supervisar y a proteger los recursos de AWS.

Temas

- [Información de seguridad de los servicios utilizados por AWS ParallelCluster](#)
- [Protección de datos en AWS ParallelCluster](#)
- [Identity and Access Management para AWS ParallelCluster](#)
- [Validación de la conformidad en AWS ParallelCluster](#)
- [Aplicación de una versión mínima de TLS 1.2](#)

Información de seguridad de los servicios utilizados por AWS ParallelCluster

- [Seguridad en Amazon EC2](#)
- [Seguridad en Amazon API Gateway](#)
- [Seguridad en AWS Batch](#)
- [Seguridad en AWS CloudFormation](#)
- [Seguridad en Amazon CloudWatch](#)
- [Seguridad en AWS CodeBuild](#)
- [Seguridad en Amazon DynamoDB](#)
- [Seguridad en Amazon ECR](#)
- [Seguridad en Amazon ECS](#)
- [Seguridad en Amazon EFS](#)
- [Seguridad en FSx para Lustre](#)
- [Seguridad en AWS Identity and Access Management \(IAM\)](#)
- [Seguridad en EC2 Image Builder](#)
- [Seguridad en AWS Lambda](#)
- [Seguridad en Amazon Route 53](#)
- [Seguridad en Amazon SNS](#)
- [Seguridad en Amazon SQS \(para AWS ParallelCluster versión 2.x.\)](#)
- [Seguridad en Amazon S3](#)
- [Seguridad en Amazon VPC](#)

Protección de datos en AWS ParallelCluster

El modelo de [responsabilidad AWS compartida modelo](#) se aplica a la protección de datos en AWS ParallelCluster. Como se describe en este modelo, AWS es responsable de proteger la infraestructura global que ejecuta 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 las [Preguntas frecuentes sobre la privacidad de datos](#). Para obtener información sobre la protección de datos en Europa, consulte la publicación de blog sobre el [Modelo de responsabilidad compartida de AWS y GDPR](#) en el Blog de seguridad de AWS .

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 multifactor (MFA) en cada cuenta.
- Utilice SSL/TLS para comunicarse con los recursos. AWS Se recomienda el uso de TLS 1.2 y recomendamos TLS 1.3.
- Configure la API y el registro de actividad de los usuarios 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 módulos criptográficos validados por FIPS 140-2 para acceder a AWS través de una interfaz de línea de comandos o una API, utilice un punto final FIPS. Para obtener más información sobre los puntos de conexión de FIPS disponibles, consulte [Estándar de procesamiento de la información federal \(FIPS\) 140-2](#).

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 AWS ParallelCluster o Servicios de AWS utiliza la consola, la API o los SDK. AWS CLI AWS 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, recomendamos encarecidamente que no incluya información de credenciales en la URL a fin de validar la solicitud para ese servidor.

Cifrado de datos

Una característica clave de cualquier servicio seguro es que la información se cifre cuando no se está utilizando activamente.

Cifrado en reposo

AWS ParallelCluster no almacena en sí mismo ningún dato del cliente aparte de las credenciales que necesita para interactuar con los AWS servicios en nombre del usuario.

En el caso de los datos de los nodos del clúster, los datos se pueden cifrar en reposo.

Para los volúmenes de Amazon EBS, el cifrado se configura mediante los ajustes [EbsSettings/Encrypted](#) y [EbsSettings/KmsKeyId](#) de la sección [EbsSettings](#). Para obtener más información, consulte el [cifrado de Amazon EBS](#) en la Guía del usuario de Amazon EC2.

Para los volúmenes de Amazon EFS, el cifrado se configura mediante los ajustes [EfsSettings/Encrypted](#) y [EfsSettings/KmsKeyId](#) de la sección de [EfsSettings](#). Para obtener más información, consulte [Cómo funciona el cifrado en reposo](#) en la Guía del usuario de Amazon Elastic File System.

En los sistemas de archivos FSx para Lustre, el cifrado de los datos en reposo se activa de forma automática al crear un sistema de archivos Amazon FSx. Para obtener más información, consulte [Cifrado de datos en reposo](#) en la Guía del usuario de Amazon FSx para Lustre.

En tipos de instancia con volúmenes NVMe, los datos incluidos en los volúmenes de almacén de instancias de NVMe se cifran con un cifrado XTS-AES-256 en un módulo de hardware de la instancia. Las claves de cifrado se generan mediante el módulo de hardware y son únicas para cada dispositivo de almacenamiento de instancias de NVMe. Todas las claves de cifrado se destruyen cuando se detiene o termina la instancia y no se pueden recuperar. No puede deshabilitar este cifrado ni tampoco proporcionar su propia clave de cifrado. Para obtener más información, consulte [Encryption at rest](#) en la Guía del usuario de Amazon EC2.

Si utiliza AWS ParallelCluster un AWS servicio que transmite los datos de los clientes a su ordenador local para su almacenamiento, consulte el capítulo sobre seguridad y conformidad de la guía del usuario de ese servicio para obtener información sobre cómo se almacenan, protegen y cifran esos datos.

Cifrado en tránsito

De forma predeterminada, todos los datos que se transmiten desde el equipo cliente en ejecución AWS ParallelCluster y desde los puntos de conexión del AWS servicio se cifran enviándolos a través de una conexión HTTPS/TLS. El tráfico entre los nodos del clúster se puede cifrar automáticamente, según los tipos de instancias seleccionados. Para obtener más información, consulte [Cifrado en tránsito](#) en la Guía del usuario de Amazon EC2.

Véase también

- [Protección de los datos en Amazon EC2](#)
- [Protección de datos en Generador de imágenes de EC2](#)
- [Protección de datos en AWS CloudFormation](#)
- [Protección de datos en Amazon EFS](#)
- [Protección de los datos en Amazon S3](#)
- [Protección de datos en FSx para Lustre](#)

Identity and Access Management para AWS ParallelCluster

AWS ParallelCluster utiliza los mismos usuarios y roles para acceder a los recursos de AWS y sus servicios. Las políticas de instancia y usuario que se AWS ParallelCluster utiliza para conceder permisos se documentan en [AWS Identity and Access Management permisos en AWS ParallelCluster](#).

La única diferencia importante es cómo se realiza la autenticación cuando se utiliza un usuario de estándar y credenciales a largo plazo. Aunque los usuarios de AWS estándar necesitan una contraseña para acceder a la consola de un servicio de , ese mismo usuario de necesita un par de claves de acceso para realizar las mismas operaciones a través de AWS ParallelCluster. Todas las demás credenciales a corto plazo se utilizan de la misma manera que con la consola.

Las credenciales que se utilizan en AWS ParallelCluster se almacenan en archivos de texto sin formato y no se cifran.

- El archivo `$HOME/.aws/credentials` almacena las credenciales a largo plazo necesarias para acceder a los recursos de AWS. Cómo recuperar el ID de clave de acceso y la clave de acceso secreta
- Las credenciales a corto plazo, como las de los roles que se adoptan o que se utilizan para servicios de AWS IAM Identity Center, también se almacenan en las carpetas `$HOME/.aws/cli/cache` y `$HOME/.aws/sso/cache`, respectivamente.

Mitigación de riesgos

- Le recomendamos encarecidamente que configure los permisos del sistema de archivos en la carpeta \$HOME/.aws, sus subcarpetas y archivos para restringir el acceso exclusivamente a los usuarios autorizados.
- Utilice roles con credenciales temporales siempre que sea posible para reducir la posibilidad de que se produzcan daños si las credenciales se ven comprometidas. Utilice credenciales a largo plazo solo para solicitar y actualizar las credenciales a corto plazo de los roles.

Validación de la conformidad en AWS ParallelCluster

Los auditores externos evalúan la seguridad y la conformidad de los servicios de AWS en distintos programas de conformidad de AWS. Usar AWS ParallelCluster para acceder a un servicio no altera la conformidad de dicho servicio.

Para obtener una lista de los servicios de AWS en el ámbito de programas de conformidad específicos, consulte [Servicios de AWS en el ámbito del programa de conformidad](#). Para obtener información general, consulte [Programas de conformidad de AWS](#).

Puede descargar los informes de auditoría de terceros utilizando AWS Artifact. Para obtener más información, consulte [Descarga de informes en AWS Artifact](#).

Su responsabilidad de conformidad al utilizar AWS ParallelCluster se determina en función de la sensibilidad de los datos, los objetivos de cumplimiento de su empresa y la legislación y los reglamentos correspondientes. AWS proporciona los siguientes recursos para ayudar con la conformidad:

- [Guías de inicio rápido de seguridad y conformidad](#): estas guías de implementación tratan consideraciones sobre arquitectura y ofrecen pasos para implementar los entornos de referencia centrados en la seguridad y la conformidad en AWS.
- Arquitectura de las medidas de seguridad y cumplimiento de HIPAA en Amazon Web Services AWS Documento técnico: en este documento técnico, se describe cómo las empresas pueden utilizar para crear aplicaciones que cumplan los requisitos de la HIPAA.
- [Recursos de conformidad de AWS](#): este conjunto de manuales y guías podría aplicarse a su sector y ubicación.
- [Evaluación de recursos con reglas](#) en la Guía para desarrolladores de AWS Config: el servicio AWS Config evalúa en qué medida las configuraciones de sus recursos cumplen las prácticas internas, las directrices del sector y las normativas.

- [AWS Security Hub](#): este servicio de AWS proporciona una vista integral de su estado de seguridad en AWS que lo ayuda a verificar la conformidad con los estándares y las prácticas recomendadas del sector de seguridad.

Aplicación de una versión mínima de TLS 1.2

Para aumentar la seguridad al comunicarse con los servicios de AWS, debe configurar la () de modo que use TLS 1.2 o una versión posterior. Cuando se utiliza AWS ParallelCluster, Python se usa para establecer la versión de TLS.

Para asegurarse de que AWS ParallelCluster no use ninguna versión de TLS anterior a TLS 1.2, es posible que tenga que volver a compilar OpenSSL para que aplique este mínimo y, a continuación, volver a compilar Python para que use el OpenSSL recién creado.

Determinar los protocolos admitidos actualmente

Primero, cree un certificado autofirmado con el fin de usarlo para el servidor de prueba y Python SDK mediante OpenSSL.

```
$ openssl req -subj '/CN=localhost' -x509 -newkey rsa:4096 -nodes -keyout key.pem -out cert.pem -days 365
```

A continuación, cree un servidor de prueba mediante OpenSSL.

```
$ openssl s_server -key key.pem -cert cert.pem -www
```

En una nueva ventana de terminal, cree un entorno virtual e instale Python SDK.

```
$ python3 -m venv test-env
source test-env/bin/activate
pip install botocore
```

Cree un script de Python denominado `check.py` que use la biblioteca HTTP subyacente del SDK.

```
$ import urllib3
URL = 'https://localhost:4433/'
```

```
http = urllib3.PoolManager(
    ca_certs='cert.pem',
    cert_reqs='CERT_REQUIRED',
)
r = http.request('GET', URL)
print(r.data.decode('utf-8'))
```

Ejecute el nuevo script.

```
$ python check.py
```

Muestra detalles sobre la conexión realizada. Busque «Protocol:» en la salida. Si el resultado es «TLSv1.2» o posterior, el SDK predeterminado es TLS v1.2 o posterior. Si se trata de una versión anterior, debe volver a compilar OpenSSL y Python.

Sin embargo, aunque la instalación de Python se establezca de forma predeterminada en TLS v1.2 o posterior, es posible que Python renegocie a una versión anterior a TLS v1.2 si el servidor no admite TLS v1.2 o posterior. Para comprobar que Python no renegocia automáticamente a versiones anteriores, reinicie el servidor de prueba con lo siguiente.

```
$ openssl s_server -key key.pem -cert cert.pem -no_tls1_3 -no_tls1_2 -www
```

Si está utilizando una versión anterior de OpenSSL, es posible que la opción `-no_tls1_3` no esté disponible. Si este es el caso, elimine la opción, porque la versión de OpenSSL que está utilizando no admite TLS v1.3. A continuación, vuelva a ejecutar el script de Python.

```
$ python check.py
```

Si su instalación de Python no renegocia correctamente las versiones anteriores a TLS 1.2, debería recibir un error de SSL.

```
$ urllib3.exceptions.MaxRetryError: HTTPSConnectionPool(host='localhost',
port=4433): Max retries exceeded with url: / (Caused by SSLError(SSLError(1, '[SSL:
UNSUPPORTED_PROTOCOL] unsupported protocol (_ssl.c:1108)')))
```

Si puede establecer una conexión, debe volver a compilar OpenSSL y Python para deshabilitar la negociación de protocolos anteriores a TLS v1.2.

Compilar OpenSSL y Python

Para asegurarse de que AWS ParallelCluster no negocie a nada anterior a TLS 1.2, debe volver a compilar OpenSSL y Python. Para ello, copie el siguiente contenido con el fin de crear un script y ejecútelos.

```
#!/usr/bin/env bash
set -e

OPENSSL_VERSION="1.1.1d"
OPENSSL_PREFIX="/opt/openssl-with-min-tls1_2"
PYTHON_VERSION="3.8.1"
PYTHON_PREFIX="/opt/python-with-min-tls1_2"

curl -O "https://www.openssl.org/source/openssl-$OPENSSL_VERSION.tar.gz"
tar -xzf "openssl-$OPENSSL_VERSION.tar.gz"
cd openssl-$OPENSSL_VERSION
./config --prefix=$OPENSSL_PREFIX no-ssl3 no-tls1 no-tls1_1 no-shared
make > /dev/null
sudo make install_sw > /dev/null

cd /tmp
curl -O "https://www.python.org/ftp/python/$PYTHON_VERSION/Python-$PYTHON_VERSION.tgz"
tar -xzf "Python-$PYTHON_VERSION.tgz"
cd Python-$PYTHON_VERSION
./configure --prefix=$PYTHON_PREFIX --with-openssl=$OPENSSL_PREFIX --disable-shared > /dev/null
make > /dev/null
sudo make install > /dev/null
```

Se compila una versión de Python que tiene un OpenSSL enlazado estáticamente que no negocia automáticamente a nada anterior a TLS 1.2. Esto también instala OpenSSL en el directorio `/opt/openssl-with-min-tls1_2` directorio e instala Python en el directorio `/opt/python-with-min-tls1_2`. Después de ejecutar este script, confirme la instalación de la nueva versión de Python.

```
$ /opt/python-with-min-tls1_2/bin/python3 --version
```

Se debería imprimir lo siguiente.

Python 3.8.1

Para confirmar que esta nueva versión de Python no negocia a una versión anterior de TLS 1.2, vuelva a ejecutar los pasos de [Determinar los protocolos admitidos actualmente](#) usando la versión de Python recién instalada (es decir, `/opt/python-with-min-tls1_2/bin/python3`).

Notas de la versión e historial de revisión

En la siguiente tabla se describen las principales actualizaciones y nuevas características de la Guía del usuario de AWS ParallelCluster . Actualizamos la documentación con frecuencia para dar respuesta a los comentarios que se nos envía.

Cambio	Descripción	Fecha
AWS ParallelCluster Publicada la versión 3.9.2	<p>Nos complace anunciar el lanzamiento de la 3.9.2 AWS ParallelCluster</p> <p>Características:</p> <ul style="list-style-type: none"> • Actualice Slurm a la 23.11.7 (desde la 23.11.4). • Para obtener más información, consulte la siguiente. CHANGELOG 3.9.2 GitHub 	28 de mayo de 2024
AWS ParallelCluster Publicada la versión 2024.05.0 de la interfaz de usuario	<p>AWS ParallelCluster Publicada la versión 2024.05.0 de la interfaz de usuario.</p> <p>Correcciones de errores:</p> <ul style="list-style-type: none"> • Se ha corregido un error en la interfaz que bloqueaba la interfaz de usuario cuando el usuario abría el panel Job Status. • Registro de cambios completo 	14 de mayo de 2024
AWS ParallelCluster Publicada la versión 2024.04.0 de la interfaz de usuario	<p>AWS ParallelCluster Publicada la versión 2024.04.0 de la interfaz de usuario.</p>	17 de abril de 2024

Características:

- Se agregó soporte para la versión 3.9.1 AWS ParallelCluster
- [Registro de cambios completo](#)

[AWS ParallelCluster Publicada la versión 3.9.1](#)

Nos complace anunciar el lanzamiento de la 3.9.1 AWS ParallelCluster

11 de abril de 2024

Para actualizar, introduce lo siguiente: `sudo pip install --upgrade aws-parallelcluster`

Correcciones de errores

- Elimine la eliminación recursiva de `mountdir` del almacenamiento compartido o al desmontar sistemas de archivos como parte de la operación de actualización del clúster.

[AWS ParallelCluster Publicada la versión 3.9.1](#)

Nos complace anunciar el lanzamiento de la 3.9.1 AWS ParallelCluster

11 de abril de 2024

Para actualizar, introduce lo siguiente: `sudo pip install --upgrade aws-parallelcluster`

Correcciones de errores

- Elimine la eliminación recursiva de `mountdir` del almacenamiento compartido o al desmontar sistemas de archivos como parte de la operación de actualización del clúster.

[AWS ParallelCluster Publicada la versión 2024.03.0 de la interfaz de usuario](#)

AWS ParallelCluster Publicada la versión 2024.03.0 de la interfaz de usuario. 12 de marzo de 2024

Características:

- Se agregó soporte para la versión 3.9.0 AWS ParallelCluster
- Se agregó soporte para Ubuntu 22.04 y Red Hat Enterprise Linux 9
- Ubuntu 18.04 obsoleto

Correcciones de errores

- Se ha corregido un problema que provocaba que algunos clústeres no aparecieran cuando se utilizaban muchos clústeres

Para obtener más información sobre los cambios, consulte los CHANGELOG archivos del [aws-parallelcluster-ui](#) paquete en cuestión GitHub.

[AWS ParallelCluster Publicada la versión 3.9.0](#)

Nos complace anunciar el lanzamiento de la 3.9.0 AWS ParallelCluster

5 de marzo de 2024

Para actualizar, introduce lo siguiente: `sudo pip install --upgrade aws-parallelcluster`

Mejoras:

- Agregue el parámetro de configuración `DeploymentSettings/DefaultUserHome` para permitir a los usuarios mover el directorio principal del usuario predeterminado al directorio principal del usuario `/local/home` en lugar de `/home` (predeterminado).
- Permite actualizar `MinCount Queue` y `ComputeResource` configurar los parámetros sin necesidad de detener la flota de cómputo. `MaxCount` Ahora es posible actualizarlos `Scheduling/SlurmSettings/QueueUpdateStrategy` configurándolos en `TERMINATE`. AWS ParallelCluster finalizará solo los nodos eliminados durante un cambio de

tamaño de la capacidad del clúster realizado mediante una actualización del clúster.

- Permite actualizar el almacenamiento compartido o externo del tipo Efs, FsxLustre FsxOntap, FsxOpenZfs y FileCache sin reemplazar la flota de procesamiento e inicio de sesión.
- Agregue soporte para RHEL9.
- Añada soporte para Rocky Linux 9 tal como CustomAmi fue creado mediante build-image el proceso. En este momento no hay ninguna AMI pública oficial de AWS ParallelCluster Rocky9 para Linux.
- Quitarla CommunicationParameters de la lista de rechazados de Slurm la configuración personalizada.
- Agregue un DeploymentSettings/DisableSudoAccessForDefaultUser parámetro para deshabilitar el acceso sudo del usuario predeterminado en los sistemas operativos compatibles.

- Cambios en los sistemas de archivos FSx for Lustre ParallelCluster creados por: Cambie la versión del servidor Lustre a 2.15.
- Añada la posibilidad de elegir entre controladores Nvidia de código abierto y cerrado al crear una AMI, mediante el atributo de nodo del ['cluster']['nvidia']['kernel_open'] libro de cocina.
- * Añada una opción de configuración clustermgtd ec2_instance_missing_max_count para permitir una cantidad configurable de reintentos para que, eventualmente, EC2 describa la coherencia de las instancias con las instancias en ejecución.

Cambios

- Actualice Slurm a la versión 23.11.4 (desde la versión 23.02.7).
- Actualice el controlador NVIDIA a la versión 535.154.05.
- Agregue soporte para Python 3.11, 3.12 en

pcluster CLI y. aws-paral-
lelcluster-batch-cli

- Cree interfaces de red utilizando el índice de tarjetas de red de la NetworkCardIndex lista de DescribeInstances respuestas de EC2, en lugar de sobrepasar el rango. MaximumNetworkCards
- No se puede crear el clúster cuando se utilizan los tipos de instancia P3, G3, P2 y G2 porque su arquitectura de GPU no es compatible con los controladores de código abierto de Nvidia (OpenRM) introducidos como parte de la versión 3.8.0.
- Actualice las dependencias de libros de cocina de terceros: nfs-5.1.2 (desde nfs-5.0.0)
- Actualice el instalador de EFA a 1.30.0.
 - Efa-driver: efa-2.6.0-1
 - Configuración EFA: efa-config-1.15-1
 - Perfil EFA: efa-profile-1.6-1
 - LibFabric-AWS: libfabric-aws-1.19.0

- Núcleo RDMA: `rdma-core-46.0-1`
- Abra MPI: `y openmpi40-aws-4.1.6-2`
`openmpi50-aws-5.0.0-11`
- Actualice NICE DCV a la versión `2023.1-16388`.
 - `server: 2023.1.16388-1`
 - `xdcv: 2023.1.565-1`
 - `gl: 2023.1.1047-1`
 - `web_viewer: 2023.1.16388-1`

Correcciones de errores

- Se solucionó el problema que hacía que el trabajo fallara cuando se enviaba como usuario de Active Directory desde los nodos de inicio de sesión. El problema se debía a una configuración incompleta de la integración con el Active Directory externo del nodo principal.
- Refactoriza las políticas de IAM definidas en la CloudFormation plantilla `parallelcluster-policies.yaml` para evitar errores en el despliegue de la API provocados por políticas

que superen los límites de IAM. ParallelCluster

- Se solucionó el problema que provocaba que los nodos de inicio de sesión no se arrancaran cuando el nodo principal tardaba más tiempo del esperado en escribir las claves.

Para obtener más información sobre los cambios, consulte los CHANGELOG archivos del paquete [aws-parallelcluster-ui](#) en. GitHub

[AWS ParallelCluster Publicada la versión 2024.02.0 de la interfaz de usuario](#)

AWS ParallelCluster Publicada la versión 2024.02.0 de la interfaz de usuario 8 de febrero de 2024

Cambios:

- Se actualizó el entorno de ejecución de Lambda a Python v3.9

Para obtener información detallada sobre los cambios, consulte los CHANGELOG archivos del [aws-parallelcluster-ui](#) paquete en. GitHub

[AWS ParallelCluster Publicada la versión 2023.12.0 de la interfaz de usuario](#)

AWS ParallelCluster Publicada la versión 2023.12.0 de la interfaz de usuario. 21 de diciembre de 2023

Características:

- Se agregó soporte para el despliegue de la PCUI con redes privadas.
- Se agregó la posibilidad de aplicar opcionalmente un límite de permisos a cada rol de IAM creado por las infraestructuras de PCUI y PCAPI
- Se agregó la posibilidad de aplicar opcionalmente un prefijo a cada rol y política de IAM creados por la infraestructura de la PCUI y la PCAPI.
- Se agregó soporte para la ParallelCluster versión 3.8.0, sin paridad de funciones en el asistente.

Para obtener más información sobre los cambios, consulte los CHANGELOG archivos del paquete [aws-parallelcluster-ui](#) en. GitHub

[AWS ParallelCluster publicada la versión 3.8.0](#)

AWS ParallelCluster publicada la versión 3.8.0 19 de diciembre de 2023

Mejoras:

- Añada compatibilidad con los bloques de capacidad de EC2 para ML.
- Añada compatibilidad con Rocky Linux 8 tal como se `CustomAmi` creó durante `build-image` el proceso. En este momento no hay ninguna AMI pública oficial de AWS ParallelCluster Rocky8 para Linux.
- Agregue un `Scheduling/ScalingStrategy` parámetro para controlar la estrategia de escalado de clústeres que se utilizará al lanzar instancias EC2 para los nodos de cómputo de Slurm. Los valores posibles son `all-or-nothing` ,, `greedy-all-or-nothing` , `best-effort` , `all-or-nothing` siendo el valor predeterminado.
- Agregue un `HeadNode/SharedStorageType` parámetro para utilizar el almacenamiento EFS en lugar de las exportaciones

NFS desde el volumen raíz del nodo principal para los recursos del sistema de archivos compartidos dentro del clúster: Intel ParallelCluster, Slurm y datos. /home Esta mejora reduce la carga en la red del nodo principal.

- Permite el montaje /home como almacenamiento compartido externo EFS o FSx a través de la SharedStorage sección del archivo de configuración.
- Añada un nuevo parámetro SlurmSettings/MungeKeySecretArn para permitir el uso de una clave MUNGE externa definida por el usuario de AWS Secrets Manager.
- Añade un Monitoring/Alarms/Enabled parámetro para activar Amazon CloudWatch Alarms en el clúster.
- Añada alarmas del nodo principal para supervisar las comprobaciones de estado del EC2, el uso de la CPU y el estado general del nodo principal, y agréguela al CloudWatch panel de control creado con el clúster.

- Añada compatibilidad con las asociaciones de repositorios de datos cuando se utilice `PERSISTENT_2` como `DeploymentType` para un FSx for Lustre gestionado.
 - Agregue un `Scheduling/SlurmSettings/Database/DatabaseName` parámetro para permitir a los usuarios especificar un nombre personalizado para la base de datos del servidor de bases de datos que se utilizará para la contabilidad de Slurm.
 - Establezca `InstanceType` un parámetro de configuración opcional al realizar la configuración `CapacityReservationTarget/CapacityReservationId` en el recurso informático.
 - Añada la posibilidad de especificar un prefijo para las funciones y políticas de IAM creadas por AWS ParallelCluster la API.
 - Añada la posibilidad de especificar un límite de permisos que se aplicará a las funciones y políticas de IAM creadas por la API.
- AWS ParallelCluster

Cambios

- Actualice Slurm a la versión 23.02.7 (desde la 23.02.6).
- Actualice el controlador NVIDIA a la versión 535.129.03.
- Actualice el kit de herramientas CUDA a la versión 12.2.2.
- Utilice los controladores de GPU NVIDIA de código abierto (OpenRM) como módulo del núcleo de NVIDIA para Linux en lugar del módulo de código cerrado de NVIDIA.
- Elimine la compatibilidad con el parámetro de `all_or_nothing_batch` configuración en el programa de currículum de Slurm y opte por la nueva `Scheduling/Scaling Strategy` configuración del clúster.
- Se cambió la convención de nomenclatura de las alarmas de clúster a '[cluster-name] - [component-name] - [metric]'.
- Cambie los tipos de volúmenes de EBS predeterminados en las regiones de ADC de gp2 a gp3, tanto para los

volúmenes raíz como para los adicionales.

- El límite de permisos opcional de la AWS ParallelCluster API ahora se aplica a todos los roles de IAM creados por la infraestructura de la API.
 - Actualiza el instalador de EFA a 1.29.1.
 - Efa-driver: efa-2.6.0-1
 - Configuración EFA: efa-config-1.15-1
 - Perfil EFA: efa-profile-1.5-1
 - LibFabric-AWS: libfabric-aws-1.19.0-1
 - Núcleo RDMA: rdma-core-46.0-1
 - Open MPI: openmpi40-aws-4.1.6-1
- Actualice GDRCopy a la versión 2.4 en todos los sistemas operativos compatibles, excepto en Centos 7, donde se usa la versión 2.3.1.
- Actualice a la versión 2.0-28aws-cfn-bootstrap .

- Se agregó soporte para Python 3.10 pulgadas. `aws-parallelcluster-batch-cli`

Correcciones de errores

- Corrige una configuración de escalado incoherente tras la reversión de la actualización del clúster al modificar la lista de tipos de instancias declarados en los recursos de cómputo.
- Corrige la generación de claves SSH de los usuarios al cambiar de usuario sin privilegios de root en clústeres integrados con un servidor LDAP externo mediante archivos de configuración de clústeres.
- Se corrigió la desactivación del modo de ahorro de energía de Slurm al configurarlo. `Scaledown IdleTime = -1`
- Corrige la ruta codificada al directorio de instalación de Slurm en el script de Slurm Accounting. `update_slurm_database_password.sh`

[AWS ParallelCluster Publicada
la versión 3.7.2](#)

AWS ParallelCluster publicada 25 de octubre de 2023
la versión 3.7.2.

Cambios:

- Actualice Slurm a la 23.02.6.

[AWS ParallelCluster Publicada la versión 2023.10.0 de la interfaz de usuario](#)

AWS ParallelCluster Publicada la versión 2023.10.0 de la interfaz de usuario. 20 de octubre de 2023

Características:

- Se agregó compatibilidad con ParallelCluster 3.7.2 con paridad de funciones en el asistente limitada a la caché de archivos FSx y compatibilidad de programación basada en memoria con varios tipos de instancias.

Correcciones de errores:

- Se ha corregido un problema que provocaba errores en la interfaz de usuario cuando la PCUI no tenía permisos para interactuar con Cost Explorer.

Mejoras

- Se mejoró la seguridad al reducir el TTL del token de acceso de 10 a 5 minutos.

Para obtener más información sobre los cambios, consulte los CHANGELOG archivos del

paquete [aws-parallelcluster-ui](#)
en. GitHub

[AWS ParallelCluster publicada
la versión 3.7.1](#)

AWS ParallelCluster publicada 22 de septiembre de 2023
la versión 3.7.1.

Cambios:

- Actualice Slurm a la 23.02.5 (desde la 23.02.4).
 - Actualice Pmix a 4.2.6 (desde 3.2.3).
 - Actualice libjwt a la versión 1.15.3 (desde la versión 1.12.0).
- Actualice el instalador de EFA a, solucionando el problema de escritura de datos de RDMA en 1.26.1 P5.
 - Controlador EFAefa-2.5.0-1 .:
 - Configuración EFA:. efa-config-1.15-1
 - Perfil EFA:. efa-profile-1.5-1
 - LibFabric-AWS:. libfabric-aws-1.18.2-1
 - Núcleo ERDMA:. rdma-core-46.0-1
 - Abre MPI:. openmpi40-aws-4.1.5-4

[AWS ParallelCluster Publicada la versión 3.7.0](#)

AWS ParallelCluster publicada la versión 3.7.0. 30 de agosto de 2023

Mejoras:

- Support la configuración de prioridades de nodos estáticas y dinámicas en los recursos informáticos mediante un archivo YAML de AWS ParallelCluster configuración.
- Ofrece la compatibilidad con Ubuntu 22. Las claves RSA no se admiten de forma predeterminada.
- Incorpora el ajuste de configuración de colas `JobExclusiveAllocation` para asignar los nodos de una partición exclusivamente a un único trabajo en un momento dado.
- Permita anular el `aws-parallelcluster-r-node` paquete en el momento de la creación y actualización del clúster. En el caso del nodo principal, esto se aplica a la actualización del clúster. Útil únicamente con fines de desarrollo.

- Evita que el servidor NFS se inicie en los nodos de computación.
- Ofrece compatibilidad con los nodos de inicio de sesión.
- Permite la programación basada en la memoria cuando se especifican varios tipos de instancias para un recurso de computación de Slurm.
- Ofrece compatibilidad para montar la caché de archivos de Amazon existente como almacenamiento compartido.

Cambios:

- De forma predeterminada, asigna a los nodos dinámicos de Slurm una prioridad (peso) de 1000. De este modo, Slurm puede priorizar los nodos estáticos inactivos sobre los nodos dinámicos inactivos.
- Haga que los `aws-parallelcluster-node` daemons solo AWS ParallelCluster manejen las particiones administradas. Slurm

- Aumenta el intervalo de sondeo de watchdog de EFS-utils a 10 segundos. Este cambio se aplica cuando EncryptionInTransit se establece en true, que es la única condición que hace que el watchdog se ejecute.
- Actualiza el instalador de EFA a 1.25.1.
 - Efa-driver: efa-2.5.0-1 (desde efa-2.1.1g)
 - Efa-config: efa-config-1.15-1 (desde efa-config-1.13-1)
 - Efa-profile: efa-profile-1.5-1 (sin cambios)
 - Libfabric-aws: libfabric-aws-1.18.1-0 (desde libfabric-aws-1.17.1-1)
 - Rdma-core: rdma-core-46.0-1 (desde rdma-core-43.0-1)
 - Open MPI: openmpi40-aws-4.1.5-4 (desde openmpi40-aws-4.1.5-1)
- Actualiza Slurm a la versión 23.02.4.

- Cambie el valor predeterminado de `Imds/` de la `v1.0` `ImdsSupport` a la `v2.0`.
- Descarta Ubuntu 18.
- Actualiza el tamaño predeterminado del volumen raíz a 40 GB para tener en cuenta los límites de CentOS 7.
- Restringe los permisos del archivo `/tmp/wait_condition_handle.txt` dentro del nodo principal para que solo el usuario raíz pueda leerlo.
- Crea un archivo JSON de asignación de `partition-nodelist` de Slurm para que los daemons del paquete de nodos lo utilicen para reconocer las particiones y listas de nodos de Slurm administradas por PC.
- Actualiza el controlador NVIDIA a la versión `535.54.03`.
- Actualiza la biblioteca CUDA a la versión `12.2.0`.
- Actualice NVIDIA Fabric Manager a `nvidia-fabricmanager-535`.
- Actualiza ARM PL a la versión `23.04.1` solo para Ubuntu 22.04.
- Actualiza NICE DCV a la versión `2023.0-15487`.

- Servidor: 2023.0.15487-1
- xdcv: 2023.0.551-1
- gl: 2023.0.1039-1
- web_viewer: 2023.0.15487-1

Correcciones de errores:

- Incorpora la validación del valor Scaledown IdleTime para evitar que se establezca un valor inferior a -1.
- Corrige un error de creación de clústeres con la AMI de aprendizaje profundo de Ubuntu en instancias de GPU con DCV habilitado.
- Se ha solucionado el problema que provocaba que se crearan políticas de IAM pendientes al crear un proveedor de recursos personalizado con ParallelCluster CloudFormation CustomLambdaRole
- Corrige un problema que provoca una mala alineación del nombre de DNS de los nodos de computación en instancias con varias interfaces de red cuando SlurmSettings/Dns/

UseEc2Hostnames se establecía en True.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes aws-parallelcluster, aws-parallelcluster-cookbook y aws-parallelcluster-node en. GitHub](#)

[Publicación de la documentación únicamente](#)

AWS ParallelCluster Se ha publicado la guía de usuario específica de la versión 3.

17 de julio de 2023

Publicación de la documentación únicamente:

- AWS ParallelCluster la versión 3 tiene su propia guía de usuario independiente.

[AWS ParallelCluster publicada la versión 3.6.1](#)

AWS ParallelCluster publicada la versión 3.6.1. 5 de julio de 2023

Cambios:

- Evita la duplicación de nodos en `clustermgtd` si se agregan nodos de computación a varias particiones de Slurm.

Correcciones de errores:

- Elimina la codificación rígida del nombre del dispositivo del volumen raíz (`/dev/sda1` y `/dev/xvda`) y la recupera de las AMI utilizadas durante `create-cluster`.
- Se corrigió el error de creación del clúster cuando se CloudFormation utilizaba un recurso personalizado con el valor `ElasticIp` establecido en `True`.
- Corrija los errores de creación y actualización del clúster cuando se utiliza un recurso AWS CloudFormation personalizado con archivos de configuración de gran tamaño.

- Corrige un problema que impedía deshabilitar la protección `ptrace` en Ubuntu y que no permitía la conexión cruzada de memoria (CMA) en `libfabric`.
- Corrige una lógica rápida de conmutación por error con una capacidad insuficiente cuando se utilizan varios tipos de instancias y no se devuelve ninguna instancia.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes `aws-parallelcluster`, `aws-parallelcluster-cookbook` y `aws-parallelcluster-node` en \[GitHub\]\(#\)](#)

[AWS ParallelCluster Publicada la versión 2023.06.0 de la interfaz de usuario](#)

AWS ParallelCluster Publicada 7 de junio de 2023
la versión 2023.06.0 de la interfaz de usuario.

Cambios:

- Se actualizó la versión predeterminada de la AWS ParallelCluster API a la 3.6.0.

Correcciones de errores:

- Se corrigió un problema de implementación en la región AWS GovCloud (EE. UU.-Oeste).
- El panel dividido ahora carga correctamente los detalles del clúster una vez iniciada la creación.

Notas:

- La función de supervisión de costes no está disponible en AWS GovCloud (US) Regions.

Para obtener más información sobre los cambios, consulte los CHANGELOG archivos del paquete [aws-parallelcluster-ui](#) en GitHub

[AWS ParallelCluster publicada la versión 3.6.0](#)

AWS ParallelCluster publicada la versión 3.6.0. 22 de mayo de 2023

Documentación:

- Agrega documentación para la [API de la biblioteca Python de AWS ParallelCluster](#).

Mejoras:

- Ofrece la compatibilidad de RHEL8.
- Agregue un [recurso AWS CloudFormation personalizado](#) para crear y administrar clústeres con CloudFormation.
- Agregue soporte para [personalizar la Slurm configuración del clúster](#) en el archivo YAML AWS ParallelCluster de configuración.
- Compila Slurm con compatibilidad de LUA.
- Aumenta de 10 a 50 el límite máximo de colas por clúster. Cada cola puede tener un máximo de 50 recursos de computación. Cada clúster puede tener un máximo de

50 recursos de computación.

- Ofrece compatibilidad para especificar una secuencia de varios [scripts de acción personalizados](#) para un evento configurado en los parámetros `OnNodeStart` , `OnNodeConfigured` y `OnNodeUpdated` .
- Incorpora una nueva sección de configuración `HealthChecks /Gpu`, para aplicar las comprobaciones de estado de la GPU en un nodo de computación antes de ejecutar un trabajo.
- Ofrece compatibilidad con `Tags` en la configuración de `SlurmQueues` y `SlurmQueues /ComputeResources` .
- Ofrece compatibilidad con [DetailedMonitoring](#) _ en la configuración de `Monitoring` .
- Agregue `mem_used_percent` `disk_used_percent` métricas para el seguimiento de la utilización del disco del volumen raíz y la memoria del nodo principal en el AWS ParallelCluster [CloudWatch panel de control](#), y configure

alarmas para supervisar estas métricas.

- Ofrece compatibilidad con la [rotación de registros](#) para los registros administrados por AWS ParallelCluster .
- Realice un seguimiento de los errores comunes de los nodos de cómputo y del tiempo de inactividad más prolongado de los nodos dinámicos en el [CloudWatch panel de control](#).
- Hace que el servidor de autenticación DCV utilice al menos un protocolo TLS-1.2 al crear el socket SSL.
- Instala el paquete [NVIDIA Data Center GPU Manager \(DCGM\)](#) en todos los sistemas operativos compatibles, excepto aarch64, centos7 y alinux2.
- Carga el módulo del kernel [nvidia-vm](#) de forma predeterminada para proporcionar la funcionalidad de memoria virtual unificada (UVM) al controlador CUDA.
- Instala [NVIDIA Persistence Daemon](#) como un servicio del sistema.

Cambios:

- Actualiza Slurm a la versión 23.02.2 (desde la versión 22.05.8).
- Actualiza munge a la versión 0.5.15 (desde la versión 0.5.14).
- Configura `TreeWidth` de Slurm en 30.
- Establece las configuraciones `prolog` y `epilog` de Slurm en el directorio de destino `/opt/slurm/etc/scripts/prolog.d/` y `/opt/slurm/etc/scripts/epilog.d/` respectivamente.
- Establece el ajuste de `BatchStartTimeout` de Slurm en un máximo de 3 minutos para ejecutar scripts `Prolog` durante el registro del nodo de computación.
- Aumente el valor predeterminado `RetentionInDays` de CloudWatch los registros de 14 a 180 días.
- Actualiza el instalador de EFA a 1.22.1.
 - Dkms: 2.8.3-2
 - Efa-driver: efa-2.1.1g (sin cambios)

- Efa-config: efa-confi
g-1.13-1 (sin cambios)
- Efa-profile: efa-profi
le-1.5-1 (sin cambios)
- Libfabric-aws:
libfabric-
aws-1.17.1-1 (desde
libfabric-aws-1.17
.0-1)
- Rdma-core: rdma-core
-43.0-1 (sin cambios)
- Open MPI: openmpi40
-aws-4.1.5-1 (sin
cambios)
- Actualiza la versión del
cliente de Lustre a 2.12
en Amazon Linux 2. El
cliente de Lustre 2.12 se ha
instalado en Ubuntu 20.04,
18.04 y CentOS >= 7.7.
- Actualiza la versión del
cliente de Lustre a 2.10.8
en CentOS 7.6.
- Actualiza el controlad
or NVIDIA a la versión
470.182.03 (desde la
versión 470.141.03).
- Actualiza NVIDIA Fabric
Manager a la versión
470.182.03 (desde la
versión 470.141.03).
- Actualiza NVIDIA CUDA
Toolkit a la versión 11.8.0
(desde la versión 11.7.1).

- Actualiza la muestra de NVIDIA CUDA a la versión 11.8.0.
- Actualiza la biblioteca MPI de Intel a la versión 2021, actualización 9 (desde la versión 2021, actualización 6). Para obtener más información, consulte [Intel® MPI Library 2021, actualización 9](#).
- Actualiza NICE DCV a la versión 2023.0-15022 (desde la versión 2022.2-14521).
 - server: 2023.0.15022-1 (desde la versión 2022.2-14521-1).
 - xdcv: 2023.0.547-1 (desde la versión 2022.2.519-1).
 - gl: 2023.0.1027-1 (desde la versión 2022.2.1012-1).
 - web_viewer: 2023.0.15022-1 (desde la versión 2022.2.14521-1).
- Actualiza aws-cfn-bootstrap a la versión 2.0-24.
- Actualice la imagen utilizada por el CodeBuild entorno al crear imágenes de contenedores para AWS Batch clústeres:

- `aws/codebuild/amazonlinux2-x86_64-standard:4.0` (desde `aws/codebuild/amazonlinux2-x86_64-standard:3.0`).
- `aws/codebuild/amazonlinux2-aarch64-standard:2.0` (desde `aws/codebuild/amazonlinux2-aarch64-standard:1.0`).

Correcciones de errores:

- Corrige los validadores de grupos de seguridad de red de Amazon EFS y Amazon FSx para evitar informar de errores falsos.
- Corrige el etiquetado que faltaba de los recursos creados por Generador de imágenes durante la operación `build-image` .
- Corrige la política de actualización de `MaxCount` para realizar siempre comparaciones numéricas en la propiedad `MaxCount` .
- Corrige la alineación de IP en las instancias de nodos de computación con varias tarjetas de red.

- Corrige la sustitución de `StoragePass` en `slurm_parallelcluster_slurmdbd.conf` cuando se realiza una actualización de los parámetros de la cola y no se actualizan las configuraciones de la contabilidad de Slurm.
- Corrige el problema que provocaba la creación de grupos de seguridad suspendidos al crear un clúster con un sistema de archivos EFS existente.
- Corrige el problema que provocaba que el daemon de `cfm-hup` fallara al reiniciarse.
- Considera los nodos dinámicos con la marca `INVALID_REG` como errores de arranque para el modo protegido de Slurm. Los nodos estáticos con error en el registro de Slurm ya se consideran errores de arranque después de `node_replacement_timeout`.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos](#)

[de los paquetes aws-paral-
lelcluster, aws-parallelcluster-
cookbook y aws-parallelcluste
r-node en. GitHub](#)

[AWS ParallelCluster Publicada la versión 2023.05.0 de la interfaz de usuario](#)

AWS ParallelCluster Publicada 16 de mayo de 2023
la versión 2023.05.0 de la interfaz de usuario.

Mejoras:

- A partir de la AWS ParallelCluster versión 3.6.0, añade soporte para RHEL 8.
- Añade el monitoreo de los costes del clúster.
- A partir de AWS ParallelCluster la versión 3.6.0, aumente las cuotas de recursos de procesamiento y colas.

Cambios:

- Se ha mejorado la interfaz de usuario del asistente de creación de clústeres.
- Se ha aumentado la velocidad de despliegue de la AWS ParallelCluster interfaz de usuario.
- Se ha mejorado la interfaz para añadir un nuevo usuario.
- De forma predeterminada, las colas se encuentran en la subred del nodo principal.

Correcciones de errores:

- Cambia a la región correcta una vez finalizada la creación del clúster.
- Corrige la visualización del indicador de carga en la característica “Editar clúster”.
- Corrige la creación de clústeres cuando se elimina la SnapshotId propiedad EBS.

Para obtener más información sobre los cambios, consulte los CHANGELOG archivos del paquete [aws-parallelcluster-ui](#) en. GitHub

[AWS ParallelCluster Publicada la versión 2023.04.0 de la interfaz de usuario](#)

AWS ParallelCluster Publicada 17 de abril de 2023
la versión 2023.04.0 de la interfaz de usuario.

Mejoras:

- Rediseño del asistente de creación de clústeres.
- Rediseño de la página de registros del clúster.
- Agrega una configuración de nombre personalizado para el almacenamiento compartido.
- Agrega una selección de almacenamiento múltiple al agregar almacenamiento a un clúster.
- Incorpora la compatibilidad de `DeletionPolicy` para Amazon EFS y FSx para Lustre.
- Agrega el ajuste `ImdsSupport` a la configuración del clúster.
- Ofrece la compatibilidad de tipos de instancias C7.
- Se ha agregado el tutorial [Volver a una versión anterior del documento de Systems Manager de AWS](#).

Cambios:

- YAML de configuración de clúster de hasta 1 MB de tamaño.
- La sesión del usuario no se ha cerrado debido a una autorización con credenciales temporales de IAM para Boto3.
- Las opciones de subprocesos múltiples están deshabilitadas cuando se selecciona una instancia de HPC.
- Se ha eliminado la opción de deshabilitar la reversión en la página de creación de clústeres.
- El usuario no puede utilizar la AWS ParallelCluster interfaz de usuario hasta que no se proporcione la información requerida.
- Se puede agregar un máximo de 10 colas.
- El documento `SSM-SessionManagerRunShell` no se sobrescribe durante la instalación de la interfaz de usuario de AWS ParallelCluster .

Correcciones de errores:

- Corrige un enlace roto para restablecer la contraseña.

- Corrige un error de delete stack causado por que EcrPrivateRepository no está vacío.
- Se ha corregido el problema de inicialización de la casilla de verificación Generar claves SSH en la sección de propiedades de administración de varios usuarios.
- Se ha corregido el bloqueo provocado por un trabajo con propiedades sin definir.
- Se ha corregido la configuración de SCRATCH FSx.
- Se ha corregido el botón de instancias de inicio y detención, que seguía habilitado después de hacer clic una vez.

Para obtener más información sobre los cambios, consulte los CHANGELOG archivos del paquete [aws-parallelcluster-ui](#) en. GitHub

[AWS ParallelCluster publicada la versión 3.5.1](#)

AWS ParallelCluster publicada la versión 3.5.1. 29 de marzo de 2023

Mejoras:

- Agrega un [instalador ejecutable](#) de la CLI de `pcluster` independiente.

Cambios:

- Actualiza el instalador de EFA a 1.22.0.
 - Efa-driver: `efa-2.1.1g` (desde `efa-2.1.1-1`)
 - Efa-config: `efa-config-1.13-1` (desde `efa-config-1.12-1`)
 - Efa-profile: `efa-profile-1.5-1` (sin cambios)
- Libfabric-aws: `libfabric-aws-1.17.0-1` (desde `libfabric-aws-1.16.1amzn3.0-1`)
- Rdma-core: `rdma-core-43.0-1` (sin cambios)
- Open MPI: `openmpi40-aws-4.1.5-1` (desde `openmpi40-aws-4.1.4-3`)

Actualiza NICE DCV a la versión 2022.2-14521 .

- `server: 2022.2.14521-1`
- `xdcv: 2022.2.519-1`
- `gl: 2022.2.1012-1`
- `web_viewer: 2022.2.14521-1`

Correcciones de errores:

- Corrige posibles errores de lanzamiento de nodos provocados por la coincidencia de patrones entre `MountDir` y `/etc/exports` al eliminar los volúmenes compartidos de Amazon EBS como parte de una actualización del clúster.
- Evita el truncamiento de los archivos de registro `compute_console_output` en cada iteración `clustermgtd`.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes `aws-parallelcluster`, `aws-parallelcluster-cookbook` y `aws-parallelcluster-node` en \[GitHub\]\(#\)](#)

[AWS ParallelCluster Publicada la versión 3.5.0](#)

AWS ParallelCluster publicada la versión 3.5.0. 20 de febrero de 2023

Mejoras:

- Permite acceder a los clústeres y administrarlos con la [interfaz de usuario de AWS ParallelCluster](#).
- Añada AWS ParallelCluster políticas versionadas en una CloudFormation plantilla a la que pueda hacer referencia en sus cargas de trabajo.
- Agrega una biblioteca de AWS ParallelCluster Python que puedas usar con tu propio código.
- Añada el registro de la salida de la consola de Compute Node a Amazon CloudWatch en caso de error de arranque del nodo de cómputo.
- Añade un campo de errores que contiene el código y el motivo del error a la salida de `describe-cluster` cuando se produce un error en la creación del clúster.
- Agrega validadores para evitar la inyección de cadenas malintencionada

al llamar al módulo de subproceso.

- No se puede crear el clúster si el estado del clúster cambia a PROTECTED cuando se aprovisionan nodos estáticos.

Cambios:

- Actualiza Slurm a la versión 22.05.8 (desde la versión 22.05.7).
- Actualiza el instalador de EFA a 1.21.0.
 - Efa-driver: efa-2.1.1-1 (desde efa-2.1)
 - Efa-config: efa-config-1.12-1 (desde efa-config-1.11-1)
 - Efa-profile: efa-profile-1.5-1 (sin cambios)
 - Libfabric-aws: libfabric-aws-1.16.1amzn3.0-1 (desde libfabric-aws-1.16.1)
 - Rdma-core: rdma-core-43.0-1 (desde rdma-core-43.0-2)
 - Open MPI: openmpi40-aws-4.1.4-3 (sin cambios)
- Hace que los registros del controlador de Slurm sean

más detallados y habilita registros adicionales para el complemento de ahorro de energía de Slurm.

Correcciones de errores:

- Corrige la creación de la base de datos del clúster verificando que el nombre del clúster no tenga más de 40 caracteres cuando la contabilidad de Slurm esté habilitada.
- Corrige un problema en `clustermgtd` que provocaba que los nodos de computación, reiniciados mediante Slurm, se sustituyeran si fallaban las comprobaciones de estado de la instancia de EC2.
- Corrige un problema que impedía el lanzamiento de nodos de computación, con reservas de capacidad compartidas por otras cuentas, debido a una política de IAM incorrecta en el nodo principal.

[Para obtener más información sobre los cambios, consulta los CHANGELOG archivos de los paquetes aws-parallelcluster, aws-parallelcluster-](#)

[cookbook, aws-parallelcluster-node y aws-parallelcluster-ui en. GitHub](#)

[AWS ParallelCluster Publicada la versión 3.4.1](#)

AWS ParallelCluster publicada la versión 3.4.1.

13 de enero de 2023

Correcciones de errores:

- Corrige un problema del programador Slurm que podía provocar una aplicación incorrecta de las actualizaciones en su registro interno de nodos de computación. Como consecuencia de este problema, las instancias de EC2 podrían dejar de estar disponibles o estar respaldadas por un tipo de instancia incorrecto.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes aws-parallelcluster, aws-parallelcluster-cookbook y aws-parallelcluster-node en. GitHub](#)

[AWS ParallelCluster Publicada la versión 3.4.0](#)

AWS ParallelCluster publicada la versión 3.4.0. 22 de diciembre de 2022

Mejoras:

- Ofrece la compatibilidad para lanzar nodos en varias zonas de disponibilidad a fin de aumentar la disponibilidad de la capacidad.
- Ofrece la compatibilidad para especificar varias subredes para cada cola a fin de aumentar la disponibilidad de la capacidad.
- Añade un nuevo parámetro de configuración en [Iam/ResourcePrefix](#) para especificar un prefijo para la ruta y el nombre de los recursos de IAM creados por AWS ParallelCluster.
- Añada una nueva sección de configuración [DeploymentSettings](#) [/LambdaFunctionsVpcConfig](#) para especificar la configuración de Vpc utilizada por las funciones de AWS ParallelCluster Lambda.
- Ofrece la posibilidad de especificar un script personalizado para que se

ejecute en el nodo principal durante una actualización del clúster. El script se puede especificar con [HeadNode/CustomActions /OnNodeUpdated](#) cuando se usa Slurm como programador.

Cambios:

- Elimina la creación de objetivos de montaje de Amazon EFS para los sistemas de archivos existentes.
- Monta los sistemas de archivos EFS mediante `amazon-efs-utils`. Los sistemas de archivos EFS se pueden montar mediante cifrado en tránsito y mediante un usuario autorizado de IAM.
- Instala stunnel 5.67 en CentOS7 y Ubuntu para admitir el cifrado en tránsito de EFS.
- Actualiza el instalador de EFA a 1.20.0 (desde 1.18.0).
 - Efa-driver: `efa-2.1` (desde `efa-1.16.0-1`)
 - Efa-config: `efa-config-1.11-1` (sin cambios)

- Efa-profile: efa-profile-1.5-1 (sin cambios)
- Libfabric-aws: libfabric-aws-1.16.1 (desde libfabric-aws-1.16.0~amzn4.0-1)
- Rdma-core: rdma-core-43.0-2 (desde rdma-core-41.0-2)
- Open MPI: openmpi40-aws-4.1.4-3 (desde openmpi40-aws-4.1.4-2)
- Actualiza Slurm a la versión 22.05.5 (desde 22.05.7).
- Actualiza Python a 3.9.16 y 3.7.16 (desde 3.9.15 y 3.7.13).
- Con Slurm 22.05.7, los nodos dinámicos en estado IDLE+CLOUD+COMPLETING+POWER_DOWN+NOT_RESPONDING no se consideran que tengan un mal estado.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes aws-parallelcluster, aws-parallelcluster-cookbook y aws-parallelcluster-node en. GitHub](#)

[AWS ParallelCluster Publicada la versión 3.3.1](#)

AWS ParallelCluster publicada la versión 3.3.1. 2 de diciembre de 2022

Cambios:

- Las AMI de AWS ParallelCluster productos oficiales ya están disponibles tras dos años de obsolescencia de Amazon EC2.
- Aumente el tamaño de la memoria de la AWS ParallelCluster API Lambda a 2048 para reducir las penalizaciones por arranque en frío y evitar tiempos de espera.

Correcciones de errores:

- Evita la sustitución de los sistemas de archivos FSx para Lustre administrados y la pérdida de datos en las actualizaciones del clúster que incluyen cambios en el identificador de subred de la flota de computación.
- La `DeletionPolicy` de [SharedStorage](#) se aplica a las acciones de actualización de clústeres.

Para obtener más información sobre los cambios, consulte el CHANGELOG archivo del paquete [aws-parallelcluster](#) en. GitHub

[AWS ParallelCluster solo documentación: hpc6id \(nota\)](#)

AWS ParallelCluster actualización solo para documentación

2 de diciembre de 2022

- AWS ParallelCluster no admite el tipo de instancia hpc6id para la configuración/.

[HeadNodeInstanceType](#)

[AWS ParallelCluster publicada la versión 3.1.5](#)

AWS ParallelCluster publicada la versión 3.1.5. 16 de noviembre de 2022

Mejoras:

- Corrige el problema de Slurm que impedía la finalización de los nodos inactivos.
- Actualiza el instalador de EFA a 1.18.0.
 - Efa-driver: efa-1.16.0-1
 - Efa-config: efa-config-1.11-1 (desde efa-config-1.9-1)
 - Efa-profile: efa-profile-1.5-1 (sin cambios)
 - Libfabric-aws: libfabric-aws-1.16.0~amzn4.0-1 (desde libfabric-1.13.2)
 - Rdma-core: rdma-core-41.0-2 (desde rdma-core-37.0)
 - Open MPI: openmpi40-aws-4.1.4-2 (desde openmpi40-aws-4.1.1-2)

Cambios:

- Añada lambda:ListTags y añada lo

lambda:UntagResource que ParallelClusterUserRole utiliza la pila de AWS ParallelCluster API para actualizar el clúster.

- Actualiza la biblioteca MPI de Intel a la versión 2021, actualización 6 (desde la versión 2021, actualización 4). Para obtener más información, consulte [Intel® MPI Library 2021, actualización 6](#).
- Actualiza el controlador NVIDIA a la versión 470.141.03 (desde 470.103.01).
- Actualiza NVIDIA Fabric Manager a la versión 470.141.03 (desde 470.103.01).

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes aws-parallelcluster, aws-parallelcluster-cookbook y aws-parallelcluster-node en. GitHub](#)

[AWS ParallelCluster Publicada la versión 3.3.0](#)

AWS ParallelCluster publicada la versión 3.3.0. 2 de noviembre de 2022

Mejoras:

- Ofrece la compatibilidad de la configuración de asignación de varias instancias para un recurso de computación cuando se utilice Slurm como programador. Para obtener más información, consulte [Asignación de varios tipos de instancias con Slurm](#).
- Ofrece la compatibilidad para agregar y eliminar [SharedStorage](#) con una actualización del clúster, utilizando una configuración actualizada. Para obtener más información, consulte [Almacenamiento compartido](#).
- Agrega un nuevo parámetro de configuración `DeletionPolicy` para los ajustes de almacenamiento compartido de [Efs](#) y [FsxLustre](#) con el objetivo de respaldar la retención del almacenamiento.
- Ofrece la compatibilidad para la contabilidad de Slurm con el nuevo

parámetro de configuración [Scheduling /SlurmSettings /Database](#). Para obtener más información, consulte [Slurmcontabilidad con AWS ParallelCluster](#).

- Ofrece la compatibilidad de reservas de capacidad bajo demanda y grupos de recursos de reserva de capacidad. Para obtener más información, consulte [Inicio de instancias con ODCR \(reservas de capacidad bajo demanda\)](#).
- Agrega un nuevo parámetro de configuración para especificar la versión de IMDS que se va a admitir en un clúster o infraestructura de creación de imágenes en el clúster, [Imds/ImdsSupport](#), y la compilación, configuraciones [Imds/ImdsSupport](#).
- Ofrece la compatibilidad de [Networking /PlacementGroup](#) en la sección [SlurmQueues /ComputeResources](#).
- Ofrece la compatibilidad para instancias con múltiples interfaces de red que estén limitadas a un solo ENI por dispositivo.

- Mejora la validación de las redes para los sistemas de archivos Amazon EFS externos comprobando el bloque CIDR en el grupo de seguridad asociado.
- Agrega un validador para comprobar si los tipos de instancias configurados admiten grupos de ubicación.
- Configura los subprocesos de NFS para que sean $\min(256, \max(8, \text{num_cores} * 4))$ a fin de garantizar una mejor estabilidad y rendimiento.
- Mueve la instalación de NFS en el momento de la compilación para reducir el tiempo de configuración.
- Habilita el cifrado del lado del servidor para el tema de EcrImageBuilder SNS que se crea al implementar la AWS ParallelCluster API y se usa para notificar los eventos de creación de imágenes de docker.

Cambios:

- Cambia el comportamiento de [SlurmQueues/Networking/Placement](#)

[Group /Enabled](#). Ahora crea un grupo de ubicación administradas único para cada recurso de computación en lugar de un único grupo de ubicaciones administradas para todos los recursos de computación.

- Ofrece la compatibilidad de [SlurmQueues /Networking /PlacementGroup /Name](#) como método de nomenclatura preferido.
- Mueve las etiquetas del nodo principal desde la plantilla de lanzamiento hasta la definición de la instancia para evitar la sustitución del nodo principal en las actualizaciones de etiquetas.
- Deshabilita el subprocesamiento múltiple mediante un script ejecutado por `cloud-init` y no a través de las `CpuOptions` establecidas en la plantilla de lanzamiento.
- Actualice Python a la versión 3.9 y NodeJS a la versión 16 en la infraestructura de API, el contenedor de Docker de la API y

los recursos Lambda del clúster.

- Elimina la compatibilidad de Python 3.6 en `aws-parallelcluster-batch-cli`.
- Actualiza Slurm a la versión 21.08.8-2 (desde 22.05.5).
- Actualiza el controlador NVIDIA a la versión 470.141.03 (desde 470.129.06).
- Actualiza NVIDIA Fabric Manager a la versión 470.141.03 (desde 470.129.06).
- Actualiza NVIDIA CUDA Toolkit a la versión 11.7.1 (from 11.4.4).
- Actualice Python utilizado en AWS ParallelCluster virtualenvs de a. 3.7.13 3.9.15
- Actualiza el instalador de EFA a la versión 1.18.0.
 - Efa-driver: efa-1.16.0-1 (sin cambios)
 - Efa-config: efa-config-1.11-1 (from efa-config-1.10-1)
 - Efa-profile: efa-profile-1.5-1 (sin cambios)

- Libfabric-aws:
`libfabric-aws-1.16.0~amzn4.0-1` (desde `libfabric-aws-1.16.0~amzn2.0-1`)
- Rdma-core: `rdma-core-41.0-2` (desde `rdma-core-37.0`)
- Open MPI: `openmpi40-aws-4.1.4-2` (desde `openmpi40-aws-4.1.1-2`)
- Actualiza NICE DCV a la versión `2022.1-13300` (desde `2022.0-12760`).
- Habilita la supresión del formulario de `SingleSubnetValidator` para `Queues`.
- No sustituye los nodos `DRAIN` cuando están en estado `COMPLETING` , ya que es posible que `Epilog` siga ejecutándose.

Correcciones de errores:

- Se corrigió el error de validación del parámetro de filtros del `AWS ParallelCluster ListClusterLogStreams` comando cuando se pasaban filtros incorrectos.

- Se corrigió el error [EfsSettings](#) `gsal` validar el parámetro [SharedStorage](#) /cuando `FileSystemId` se especificaba junto con otros [EfsSettings](#) parámetros [SharedStorage](#) /.
- Anteriormente, `FileSystemId` no estaba incluido.
- Corrige la actualización del clúster al cambiar el orden de [SharedStorage](#) junto con otros cambios en la configuración.
- Se `UpdateParallelClusterLambdaRole` corrigió en la AWS ParallelCluster API a la que se cargaban los registros CloudWatch.
- Corrige el error de que Cinc no utilizara el paquete de certificados de CA locales al instalar los paquetes antes de ejecutar cualquier libro de recetas.
- Corrige un problema al actualizar Ubuntu con `pcluster build-image` cuando se establece `Build:UpdateOsPackages:Enabled:true`.
- Corrige el error al analizar la configuración del clúster YAML al duplicar las claves.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes aws-parallelcluster, aws-parallelcluster-cookbook y aws-parallelcluster-node en. GitHub](#)

[AWS ParallelCluster solo se agregó una referencia a la API en la documentación.](#)

AWS ParallelCluster actualiza ción solo para la documentación

27 de octubre de 2022

- Se ha agregado la versión 3 [Referencia de la API de AWS ParallelCluster](#) a la documentación.

[AWS ParallelCluster publicada la versión 3.2.1](#)

AWS ParallelCluster publicada la versión 3.2.1. 3 de octubre de 2022

Mejoras:

- Mejora la lógica para asociar las tablas de enrutamiento del host a las distintas tarjetas de red a fin de admitir mejor las instancias de EC2 con varias NIC.

Cambios:

- Actualiza el controlador NVIDIA a la versión 470.141.03.
- Actualiza NVIDIA Fabric Manager a la versión 470.141.03.
- Deshabilita las tareas `man-db` y `mlocate` del trabajo `cron`, ya que pueden tener un impacto negativo en el rendimiento del nodo.
- Actualiza la biblioteca MPI de Intel a la versión 2021.6.0.602.
- Actualiza Python de la versión 3.7.10 a la 3.7.13 en respuesta a este riesgo de seguridad.

Correcciones de errores:

- Evita que se produzca un error en `DescribeCluster` cuando la configuración del clúster no esté disponible.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes `aws-parallelcluster`, `aws-parallelcluster-cookbook` y `aws-parallelcluster-node` en GitHub](#)

[AWS ParallelCluster Publicada la versión 3.2.0](#)

AWS ParallelCluster publicada la versión 3.2.0 27 de julio de 2022

Mejoras:

- Ofrece la compatibilidad de la [programación basada en memoria](#) en Slurm.
 - Configura la memoria real de los nodos de computación en la configuración del clúster de Slurm.
 - Agrega un nuevo parámetro de configuración [Scheduling /SlurmSettings /EnableMemoryBasedScheduling](#) para habilitar la programación basada en memoria en Slurm.
 - Agrega un nuevo parámetro de configuración [Scheduling /SlurmQueues /ComputeResources /SchedulableMemory](#) para anular el valor predeterminado de la memoria que ve el programador en los nodos de computación.

- Mejora la flexibilidad de las actualizaciones de configuración del clúster para evitar que todo el clúster se detenga y se inicie siempre que sea posible. Agrega un nuevo parámetro de configuración [Scheduling /SlurmSettings /QueueUpdateStrategy](#) para establecer la estrategia preferida que se utilizará cuando los nodos de computación necesiten actualizar o reemplazar la configuración.
- Mejora el mecanismo de conmutación por error con respecto a los recursos de computación disponibles cuando se produzcan problemas de capacidad insuficiente con las instancias de EC2. [Deshabilita los nodos de computación durante un periodo de tiempo configurable](#) cuando el lanzamiento de un nodo falle debido a que la capacidad es insuficiente.
- Ofrece la compatibilidad para montar los sistemas de archivos existentes [FSx for ONTAP](#) y [FSx for OpenZFS](#).

- Ofrece la compatibilidad para montar varias instancias de los sistemas de archivos [Amazon Elastic File Systems](#), [FSx para Lustre](#), [FSx para ONTAP](#) y [FSx para OpenZFS](#) existentes.
- Ofrece la compatibilidad del [tipo de implementación FSx para Lustre Persistent_2](#) al crear un nuevo sistema de archivos.
- Cuando utilice el asistente de [pcluster configure](#), pide al usuario que habilite EFA para los tipos de instancias compatibles.
- Ofrece la compatibilidad para reiniciar nodos de computación mediante Slurm.
- Mejora el manejo de los estados de energía de Slurm para tener en cuenta también el apagado manual de los nodos.
- Instala NVIDIA GDRCopy 2.3 en las AMI del producto para permitir la copia de memoria de la GPU con baja latencia.

Cambios:

- Actualiza el instalador de EFA a la versión 1.17.2.
 - Controlador de EFA: efa-1.16.0-1
 - Configuración de EFA: efa-config-1.10-1
 - Perfil de EFA: efa-profile-1.5-1
 - Libfabric: libfabric-aws-1.16.0~amzn2.0-1
 - Núcleo de RDMA: rdma-core-41.0-2
 - Open MPI: openmpi40-aws-4.1.4-2
- Actualiza NICE DCV a la versión 2022.0-12760.
- Actualiza el controlador NVIDIA a la versión 470.129.06.
- Actualiza NVIDIA Fabric Manager a la versión 470.129.06.
- Cambia los tipos de volúmenes de EBS predeterminados de gp2 a gp3 tanto en los volúmenes raíz como en los adicionales.
- Cambios en los sistemas de archivos FSx for Lustre creados por: AWS ParallelCluster

- Cambia el tipo de implementación predeterminado a `Scratch_2`.
- Cambia la versión del servidor de Lustre a `2.12`.
- No es necesario configurar [Placement Group /Enabled](#) como `true` al pasar un `PlacementGroup /Id` existente.
- No permite establecer `PlacementGroup /Id` cuando `Placement Group /Enabled` está explícitamente establecido en `false`.
- Agrega una etiqueta `parallelcluster:cluster-name` a todos los recursos creados por AWS ParallelCluster.
- Agregue `lambda:ListTags` y `ParallelClusterUserRole` utilice `lambda:UntagResource` la pila de AWS ParallelCluster API para la actualización del clúster.
- Restringe el acceso de IPv6 a IMDS únicamente para los usuarios administradores de clústeres y raíz cuando el

parámetro de configuración `HeadNode/Imds/Secured` esté habilitado.

- Con una AMI personalizada, utilice el tamaño del volumen raíz de la AMI en lugar del tamaño ParallelCluster predeterminado de 35 GiB. El valor se puede cambiar en el archivo de configuración del clúster.
- Deshabilita automáticamente la flota de computación cuando el parámetro de configuración `Scheduling /SlurmQueues /ComputeResources /SpotPrice` es menor que el precio mínimo requerido de cumplimiento de solicitud de Spot.
- Muestra los valores `requested_value` y `current_value` del conjunto de cambios al añadir o eliminar una sección durante una actualización.
- Deshabilita el servicio `aws-ubuntu-eni-helper`, disponible en las AMI de aprendizaje profundo, para evitar conflictos con `configure_nw_interface.sh` al configura

r instancias con varias tarjetas de red.

- Elimina la compatibilidad de Python 3.6.
- Establece la MTU en 9001 para todas las interfaces de red al configurar instancias con varias tarjetas de red.
- Elimina el punto final al configurar el FQDN del nodo de computación.
- Administra los nodos estáticos en `POWERING_DOWN` .
- No reemplaza el nodo dinámico en `POWER_DOWN` , ya que es posible que los trabajos sigan ejecutándose.
- Reinicia los daemons de `clustermgtd` y `slurmctld` en el momento de la actualización del clúster solo cuando se actualizan los parámetros de `Scheduling` en la configuración del clúster.
- Actualiza `slurmctld` y los archivos de mantenimiento `slurmd` y `systemd`.
- Restringe el acceso de IPv6 a IMDS únicamente para los usuarios administradores de clústeres y raíz cuando el parámetro de configuración

HeadNode/Imds/Secured esté habilitado.

- Establece la configuración de Slurm AuthInfo=cred_expire=70 para reducir el tiempo que deben esperar los trabajos en cola antes de volver a empezar cuando los nodos no estén disponibles.
- Actualiza las dependencias de libros de recetas de terceros:
 - apt-7.4.2 (desde apt-7.4.0)
 - line-4.5.2 (desde line-4.0.1)
 - openssh-2.10.3 (desde openssh-2.9.1)
 - pyenv-3.5.1 (desde pyenv-3.4.2)
 - selinux-6.0.4 (desde selinux-3.1.1)
 - yum-7.4.0 (desde yum-6.1.1)
 - yum-epel-4.5.0 (desde yum-epel-4.1.2)

Correcciones de errores:

- Corrija el comportamiento predeterminado para omitir los pasos de AWS ParallelCluster validación y prueba

al crear una AMI personalizada.

- Corrige la filtración del identificador del archivo en `computemgtd` .
- Se corrigió una condición de carrera que provocaba que, de forma esporádica, las instancias lanzadas se cancelaran inmediatamente porque aún no estaban disponibles en la respuesta de `DescribeInstances` EC2.
- Corrige la compatibilidad del parámetro `DisableSimultaneousMultithreading` en los tipos de instancias con procesadores Arm.
- Se corrigió un error en la actualización de la pila de AWS ParallelCluster API al actualizar desde una versión anterior. Agrega el patrón de recursos utilizado para la acción `ListImagePipelineImages` en el `EcrImageDeletionLambdaRole` .
- Se AWS ParallelCluster corrigió la API que añadía los permisos faltantes necesarios para importar o exportar desde Amazon

S3 al crear un sistema de archivos FSx for Lustre.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes aws-parallelcluster, aws-parallelcluster-cookbook y aws-parallelcluster-node en. GitHub](#)

[AWS ParallelCluster actualiza ciones solo de la documentación de este año hasta la fecha](#)

AWS ParallelCluster actualiza ciones exclusivas de la documentación.

6 de julio de 2022

Nuevas secciones:

- [Prácticas recomendadas: alertas de presupuesto](#) versión 3
- [Mejores prácticas: mover un clúster a una nueva versión AWS ParallelCluster secundaria o a una versión de parche](#) versión 3
- [Uso de Amazon S3](#) versión 3
- [Uso de instancias de spot](#) versión 3
- [Modo protegido de clúster Slurm](#) versión 3
- [Recursos y etiquetado de AWS ParallelCluster](#) versión 3
- [Panel de Amazon CloudWatch](#) versión 3
- [Integración con Amazon CloudWatch Logs](#) versión 3
- [Elastic Fabric Adapter](#) versión 3
- [AWS ParallelCluster Personalización de AMI](#) versión 3

- [Inicio de instancias con ODCR \(reservas de capacidad bajo demanda\)](#) versión 3
- [Parcheo de AMI y sustitución de instancias de EC2](#) versión 3
- [Cómo funciona AWS ParallelCluster](#) versión 3
- [Configuración del cifrado de almacenamiento compartido o con una clave AWS KMS](#) versión 3
- [Ejecución de trabajos en un clúster en modo de cola múltiple](#) versión 3
- [Uso de la API AWS ParallelCluster](#) versión 3

Actualizaciones de secciones:

- [Prácticas recomendadas: rendimiento de la red](#) versión 3: se han agregado las prácticas recomendadas para usar Elastic Fabric Adaptor.
- [AWS Identity and Access Management permisos en AWS ParallelCluster](#) versión 3: se han realizado varias actualizaciones y agregado [Política de usuario adicional pcluster de AWS](#)

[ParallelCluster al utilizar Amazon FSx para Lustre.](#)

- [AWS ParallelCluster solución de problemas](#)
versión 3: se han realizado varias actualizaciones.

[AWS ParallelCluster publicada la versión 3.1.4](#)

AWS ParallelCluster publicada la versión 3.1.4. 16 de mayo de 2022

Mejoras:

- Agrega la validación para [Directory Service / PasswordSecretArn](#) para que falle si el secreto no existe.

Ofrece la compatibilidad para habilitar la autenticación JWT en Slurm.

Cambios:

- Actualiza Slurm a la versión 21.08.8-2.
- Crea la compatibilidad de Slurm con JWT.
- No es necesario configurar [Placement Group / Enabled](#) como true al pasar un PlacementGroup /Id existente.
- Se añade `lambda:TagResource` a la pila de ParallelCluster API `ParallelClusterUserRole` utilizada para la creación de clústeres e imágenes.

Correcciones de errores:

- Corrige la posibilidad de exportar los registros de un clúster cuando se utilizaba el comando `export-cluster-logs` con la opción `--filters`.
- Corrija el punto de entrada de AWS Batch Docker para que utilice un directorio `/home` compartido para coordinar la ejecución de trabajos paralelos y multinodos.
- Restablece la dirección del nodo al configurar el nodo estático en mal estado de Slurm como inactivo para evitar tratar el nodo estático fallido con capacidad insuficiente como un nodo con un error de arranque.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes `aws-parallelcluster`, `aws-parallelcluster-cookbook` y `aws-parallelcluster-node` en GitHub](#)

[AWS ParallelCluster Publicada la versión 3.1.3](#)

AWS ParallelCluster publicada la versión 3.1.3. 20 de abril de 2022

Mejoras:

- Ejecuta la creación de la clave SSH junto con la creación del directorio o HOME, por ejemplo, al iniciar sesión con SSH, al cambiar a otro usuario y al ejecutar un comando como otro usuario.
- Ofrece la compatibilidad de los nombres distintivos FQDN y LDAP en el parámetro de configuración [Directory Service /DomainName](#). El nuevo validador comprueba ahora ambas sintaxis.
- El nuevo script `update_directory_service_password.sh` implementado en el nodo principal admite la actualización manual de la contraseña de Active Directory en la configuración de SSSD. El as recupera la contraseña desde AWS Secrets Manager la configuración del clúster.

- Ofrece compatibilidad para implementar la infraestructura de API en entornos sin una VPC predeterminada.

Cambios:

- Deshabilita los estados C más profundos en las AMI oficiales x86_64 y en las AMI creadas mediante el comando `build-image`, para garantizar un alto rendimiento y una baja latencia.
- Actualizaciones del paquete del sistema operativo y correcciones de seguridad.
- Cambia las imágenes base de Amazon Linux 2 para usar AMI con Kernel 5.10.

Correcciones de errores:

- Corrige la pila de imágenes de compilación en `DELETE_FAILED` tras la creación correcta de la imagen, debido a las nuevas políticas de Generador de imágenes de EC2.
- Corrige la conversión del parámetro de configuración [Directory Service](#) / [DomainAdd](#)

r a la propiedad `ldap_uri` de SSSD cuando contiene varias direcciones de dominio.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes aws-parallelcluster y aws-parallelcluster-cookbook en GitHub](#)

[AWS ParallelCluster Publicada la versión 3.1.2](#)

AWS ParallelCluster publicada la versión 3.1.2. 2 de marzo de 2022

Cambios:

- Actualiza Slurm a la versión 21.08.6 (desde 21.08.5).

Correcciones de errores:

- Corrige la actualización del archivo `/etc/hosts` en los nodos de computación cuando un clúster se implementa en subredes sin acceso a Internet.
- Corrige el proceso de arranque de los nodos de computación para que esperen a que se inicialicen las unidades efímeras antes de unirse al clúster.

Para obtener más información sobre los cambios, consulte los CHANGELOG archivos del paquete [aws-parallelcluster](#) en GitHub

[AWS ParallelCluster publicada la versión 3.1.1](#)

AWS ParallelCluster publicada la versión 3.1.1. 10 de febrero de 2022

- Ofrece la compatibilidad de varios entornos de clústeres de usuarios [integrándolos con los dominios de Active Directory \(AD\)](#) administrados mediante AWS Directory Service.
- Ofrece la compatibilidad de [UseEc2Hostnames](#) en el archivo de configuración del clúster. Si se establece en verdadero, utiliza los nombres de host predeterminados de EC2 (por ejemplo, ip-1-2-3-4) para los nodos de computación.
- Ofrece compatibilidad para la creación de clústeres en [subredes sin acceso a Internet](#).
- Ofrece la compatibilidad de varios tipos de instancias de computación por cola.
- Ofrece la compatibilidad de la programación de GPU con Slurm en instancias ARM con tarjetas NVIDIA.
- Agregue indicadores abreviados para `cluster-name (-n)`, `region (-r)`, `image-id (-i)`

`ycluster-configuration /image-configuration (-c)` a la AWS ParallelCluster CLI.

- Ofrece la compatibilidad de la opción `NEW_CHANGED_DELETED` para el parámetro [AutoImportPolicy](#) de FSx para Lustre.
- Agrega una etiqueta `parallelcluster:compute-resource-name` a los recursos `LaunchTemplates` de EC2 utilizados por los nodos de computación.
- Mejora los grupos de seguridad creados dentro del clúster para permitir las conexiones entrantes desde grupos de seguridad personalizados cuando se especifiquen los parámetros `SecurityGroups` para algunos nodos principales o colas.
- Instala los controladores NVIDIA y la biblioteca CUDA para ARM.

Cambios:

- Actualiza Slurm a la versión 21.08.5 (desde 20.11.8).

- Actualiza el complemento de Slurm a la versión 21.08 (desde 20.11).
- Actualiza NICE DCV a la versión 2021.3-11591 (desde 2021.1-10851).
- Actualiza el controlador NVIDIA a la versión 470.103.01 (desde 470.57.02).
- Actualiza NVIDIA Fabric Manager a la versión 470.103.01 (desde 470.57.02).
- Actualiza CUDA a la versión 11.4.4 (desde 11.4.0).
- [Intel MPI](#) se ha actualizado a la versión 2021, actualización 4 (actualizado desde la versión 2019, actualización 8). Para obtener más información, consulte [Intel® MPI Library 2021, actualización 4](#).
- Actualiza PMIx a la versión 3.2.3 (desde 3.1.5).
- Elimina el volcado de nodos de computación fallidos a `/home/logs/compute` . Los archivos de registro de los nodos de cómputo están disponibles en CloudWatch y dentro de los registros de la consola de EC2.

- Habilita la posibilidad de suprimir los validadores de longitud `SlurmQueues` y `ComputeResources`.
- Deshabilita la actualización del paquete en el momento del lanzamiento de la instancia en Amazon Linux 2.
- Deshabilite los metadatos de imagen ImageBuilder mejorados de EC2 al crear imágenes AWS ParallelCluster personalizadas.
- Establece explícitamente que la fuente de datos `cloud-init` sea EC2. Esto ahorra tiempo de arranque para las plataformas Ubuntu y CentOS.
- Use el nombre del recurso de computación en lugar del tipo de instancia en el nombre de la plantilla de lanzamiento de la flota de computación.
- Redirige `stderr` y `stdout` al archivo de registro de CLI para evitar que aparezca texto no deseado en la salida de CLI del `pcluster`.
- Mueve las recetas de configuración/instalación para separar los libros de recetas a los que se llama

desde el libro principal
. Los puntos de entrada
existentes se mantienen
y son compatibles con
versiones anteriores.

- Descarga las dependencias de la plataforma Intel HPC durante el tiempo de creación de AMI para evitar el contacto con Internet durante la creación del clúster.
- No elimina - del nombre del recurso de computación al configurar los nodos de Slurm.
- No configura las GPU en Slurm cuando el controlador NVIDIA no está instalado.
- Corrige el permiso `ecs:ListContainerInstances` en `BatchUserRole` .
- Corrige la exportación de los registros del clúster cuando no se especifica ningún prefijo, previamente exportados a un prefijo `None`.
- Corrige el error de no realizar la reversión cuando se producía un error en la actualización del clúster.
- Corrige el permiso `ecs:ListContainerI`

instances en BatchUser
Role .

- Corrige el esquema RootVolume para HeadNode generando un error si se especifica un KmsKeyId no compatible.
- Corrige las métricas faltantes de Amazon FSx para que se muestren en CloudWatch el panel de control.
- Corrige EfaSecurityGroupValidator . Anteriormente, era posible que se produjeran errores falsos cuando se proporcionaban grupos de seguridad personalizados y se habilitaba la EFA.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes aws-parallelcluster, aws-parallelcluster-cookbook y aws-parallelcluster-node en. GitHub](#)

[AWS ParallelCluster Publicada la versión 3.0.3](#)

AWS ParallelCluster publicada la versión 3.0.3. 17 de enero de 2022

- Deshabilita el agente de `log4j-cve-2021-44228-hotpatch` (`Log4jHotPatch`) en Amazon Linux 2 para evitar una posible degradación del rendimiento. Para obtener más información, consulte el [anuncio del hotpatch de Amazon Linux para Apache Log4j](#).

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes `aws-parallelcluster` y `aws-parallelcluster-cookbook` en \[GitHub\]\(#\)](#)

[AWS ParallelCluster Publicada la versión 3.0.2](#)

AWS ParallelCluster publicada la versión 3.0.2. 5 de noviembre de 2021

Actualiza el instalador de [Elastic Fabric Adapter](#) a 1.14.1.

- Configuración de EFA: efa-config-1.9-1 (desde efa-config-1.9)
- Perfil de EFA: efa-profile-1.5-1 (desde efa-profile-1.5)
- Módulo Kernel de EFA: efa-1.14.2 (desde efa-1.13.0)
- Núcleo de RDMA: rdma-core-37.0 (desde rdma-core-35)
- Libfabric: libfabric-1.13.2 (desde libfabric-1.13.0)
- Open MPI: openmpi40-aws-4.1.1-2 (sin cambios)

El RDMA de GPUDirect siempre está habilitado si el tipo de instancia lo admite. La opción [GdrSupport](#) de configuración no tiene ningún efecto.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes aws-parallelcluster, aws-parallelcluster-cookbook y aws-parallelcluster-node en. GitHub](#)

[AWS ParallelCluster Publicada la versión 3.0.1](#)

AWS ParallelCluster publicada la versión 3.0.1 27 de octubre de 2021

Herramienta de migración de la configuración de clústeres

- Los clientes ahora pueden migrar las configuraciones de sus clústeres del formato de la AWS ParallelCluster versión 2 al formato de la AWS ParallelCluster versión 3 basado en YAML. Para obtener más información, consulte [pcluster3-config-converter](#).

El nodo principal se puede detener

- [Tras detener la flota de cómputo, se puede detener el nodo principal y reiniciar lo posteriormente mediante la consola Amazon EC2 o el comando stop-instances.](#)

AWS CLI

`~/.aws/config` Lectura predeterminada del archivo Región de AWS

- Para el [pcluster](#) comando, si no Región de AWS se especifica en el

archivo de configuración, en el entorno o en la línea de comandos, se utiliza el valor predeterminado Región de AWS especificado en la `region` configuración de la `[default]` sección del `~/.aws/config` archivo.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes `aws-parallelcluster`, `aws-parallelcluster-cookbook` y `aws-parallelcluster-node` en \[GitHub\]\(#\)](#)

[AWS ParallelCluster Publicada la versión 3.0.0](#)

AWS ParallelCluster publicada la versión 3.0.0. 10 de septiembre de 2021

Compatibilidad de la administración de clústeres mediante Amazon API Gateway

- Los clientes ahora pueden administrar e implementar clústeres a través de puntos de conexión HTTP con Amazon API Gateway. Esto abre nuevas posibilidades para los flujos de trabajo con scripts o basados en eventos.

La interfaz de línea de AWS ParallelCluster comandos (CLI) también se ha rediseñado para que sea compatible con esta API e incluye una nueva opción de salida JSON. Esta nueva funcionalidad permite a los clientes implementar capacidades de componentes básicos similares también mediante la CLI.

Creación de AMI personalizadas mejorada

- Los clientes ahora tienen acceso a un proceso más sólido para crear y administrar AMI personalizadas mediante Generador de imágenes de EC2. Las AMI personalizadas ahora se pueden administrar a través de un archivo de AWS ParallelCluster configuración independiente y se pueden crear mediante el [pcluster build-image](#) comando de la interfaz de línea de AWS ParallelCluster comandos.

[Para obtener más información sobre los cambios, consulte los CHANGELOG archivos de los paquetes aws-parallelcluster, aws-parallelcluster-cookbook y aws-parallelcluster-node en. GitHub](#)

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