aws

Benutzer-Leitfaden

AWS STK.

AWS STK.: Benutzer-Leitfaden

Table of Contents

Was ist AWS Parallel Computing Service?

AWS Parallel Computing Service (AWS PCS) ist ein verwalteter Service, der es einfacher macht, HPC-Workloads (High Performance Computing) auszuführen und zu skalieren und wissenschaftliche und technische Modelle für die AWS Verwendung von Slurm zu erstellen. Verwenden Sie AWS PCS, um Rechencluster aufzubauen, die erstklassige AWS Rechen-, Speicher-, Netzwerk- und Visualisierungsfunktionen integrieren. Führen Sie Simulationen durch oder erstellen Sie wissenschaftliche und technische Modelle. Rationalisieren und vereinfachen Sie Ihren Clusterbetrieb mithilfe der integrierten Management- und Observability-Funktionen. Geben Sie Ihren Benutzern die Möglichkeit, sich auf Forschung und Innovation zu konzentrieren, indem Sie ihnen ermöglichen, ihre Anwendungen und Jobs in einer vertrauten Umgebung auszuführen.

Themen

Konzepte in AWS PCS

Ein Cluster in AWS PCS hat eine oder mehrere Warteschlangen, die mindestens einer Rechenknotengruppe zugeordnet sind. Jobs werden an Warteschlangen weitergeleitet und auf EC2 Instanzen ausgeführt, die durch Rechenknotengruppen definiert sind. Sie können diese Grundlagen nutzen, um anspruchsvolle HPC-Architekturen zu implementieren.

Cluster

Ein Cluster ist eine Ressource für die Verwaltung von Ressourcen und die Ausführung von Workloads. Ein Cluster ist eine AWS PCS-Ressource, die eine Zusammenstellung von Rechen-, Netzwerk-, Speicher-, Identitäts- und Job-Scheduler-Konfigurationen definiert. Sie erstellen einen Cluster, indem Sie angeben, welchen Job-Scheduler Sie verwenden möchten (derzeit Slurm), welche Scheduler-Konfiguration Sie wünschen, welchen Service Controller Sie für die Verwaltung des Clusters verwenden möchten und in welcher VPC die Cluster-Ressourcen gestartet werden sollen. Der Scheduler akzeptiert und plant Jobs und startet auch die Rechenknoten (EC2 Instanzen), die diese Jobs verarbeiten.

Compute-Knotengruppe

Eine Rechenknotengruppe ist eine Sammlung von Rechenknoten, die AWS PCS verwendet, um Jobs auszuführen oder interaktiven Zugriff auf einen Cluster zu ermöglichen. Wenn Sie eine Compute-

Knotengruppe definieren, geben Sie allgemeine Merkmale wie EC2 Amazon-Instance-Typen, minimale und maximale Instance-Anzahl, Ziel-VPC-Subnetze, Amazon Machine Image (AMI), Kaufoption und benutzerdefinierte Startkonfiguration an. AWS PCS verwendet diese Einstellungen, um Rechenknoten in einer Rechenknotengruppe effizient zu starten, zu verwalten und zu beenden.

Warteschlange

Wenn Sie einen Job auf einem bestimmten Cluster ausführen möchten, senden Sie ihn an eine bestimmte Warteschlange (manchmal auch Partition genannt). Der Job bleibt in der Warteschlange, bis AWS PCS plant, dass er auf einer Rechenknotengruppe ausgeführt wird. Sie ordnen jeder Warteschlange eine oder mehrere Rechenknotengruppen zu. Eine Warteschlange ist erforderlich, um Jobs auf den zugrunde liegenden Compute-Knotengruppenressourcen unter Verwendung verschiedener vom Job-Scheduler angebotener Planungsrichtlinien zu planen und auszuführen. Benutzer reichen Jobs nicht direkt an einen Rechenknoten oder eine Rechenknotengruppe weiter.

Systemadministrator

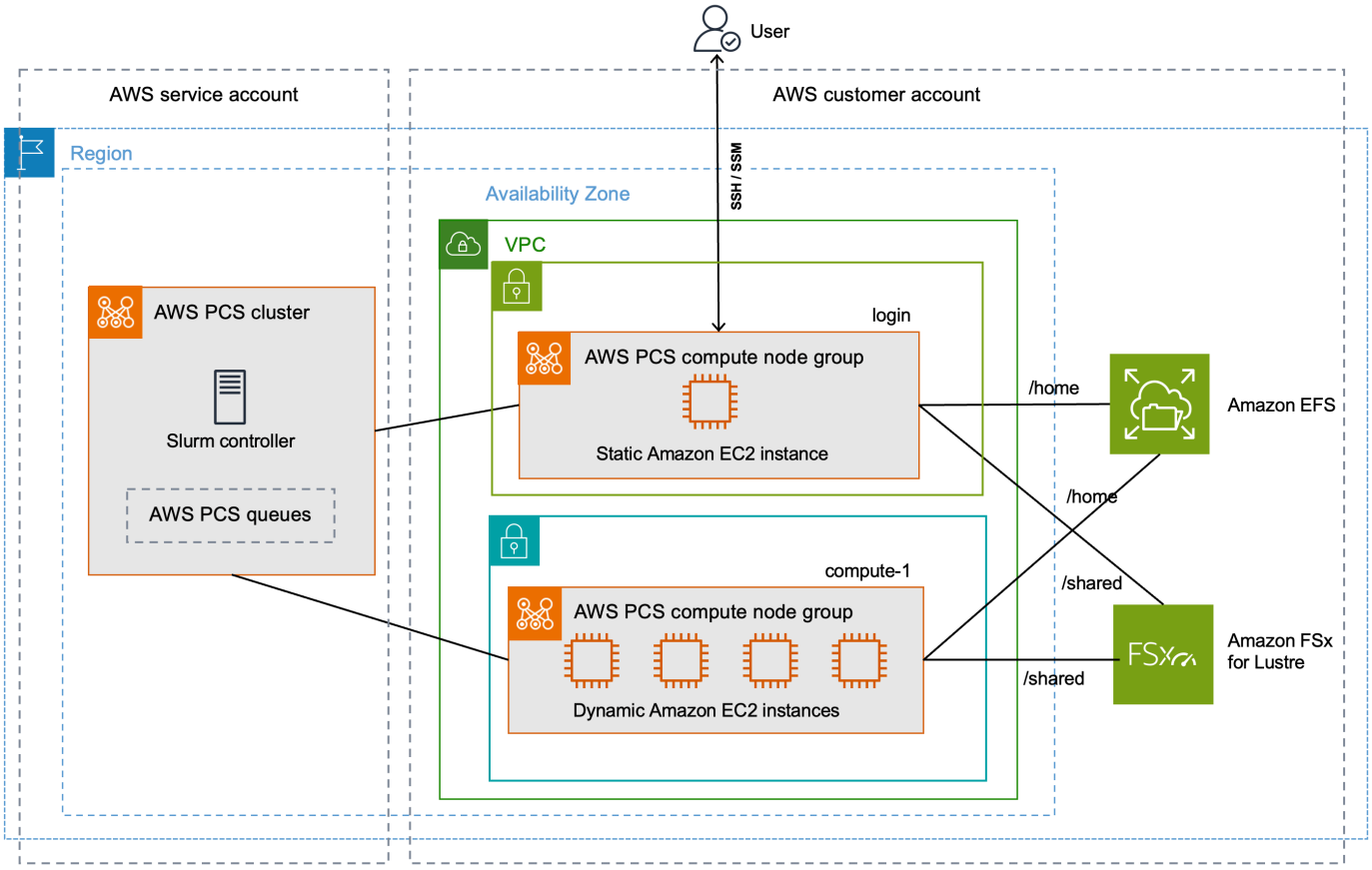
Ein Systemadministrator stellt einen Cluster bereit, verwaltet und betreibt ihn. Sie können über die AWS PCS API und AWS-Managementkonsole das AWS SDK auf AWS PCS zugreifen. Sie haben über SSH Zugriff auf bestimmte Cluster oder können dort administrative Aufgaben ausführen AWS Systems Manager, Jobs ausführen, Daten verwalten und andere Shell-basierte Aktivitäten ausführen. Weitere Informationen finden Sie in der AWS Systems Manager Dokumentation.

Endbenutzer

Ein Endbenutzer ist nicht dafür day-to-day verantwortlich, einen Cluster bereitzustellen oder zu betreiben. Sie verwenden eine Terminalschnittstelle (wie SSH), um auf Clusterressourcen zuzugreifen, Jobs auszuführen, Daten zu verwalten und andere Shell-basierte Aktivitäten durchzuführen.

Erste Schritte mit AWS Parallel Computing Service

Dies ist ein Tutorial zum Erstellen eines einfachen Clusters, mit dem Sie AWS PCS testen können. Die folgende Abbildung zeigt das Design des Clusters.



Das Cluster-Design des Tutorials besteht aus den folgenden Hauptkomponenten:

- Eine VPC und Subnetze, die die AWS PCS-Netzwerkanforderungen erfüllen.
- Ein Amazon EFS-Dateisystem, das als gemeinsames Home-Verzeichnis verwendet wird.
- Ein Amazon FSx for Lustre-Dateisystem, das ein gemeinsam genutztes Hochleistungsverzeichnis bereitstellt.
- Ein AWS PCS-Cluster, der einen Slurm-Controller bereitstellt.
- 2 AWS PCS-Rechenknotengruppen.
  - Die `login` Knotengruppe, die einen Shell-basierten interaktiven Zugriff auf das System ermöglicht.
  - Die `compute-1` Knotengruppe bietet elastisch skalierbare Instanzen zur Ausführung von Jobs.

- 1 Warteschlange, die Jobs an EC2 Instanzen in der `compute-1` Knotengruppe sendet.

Der Cluster benötigt zusätzliche AWS Ressourcen wie Sicherheitsgruppen, IAM-Rollen und EC2 Startvorlagen, die im Diagramm nicht dargestellt sind.

> **Note**
>
> Wir empfehlen, dass Sie die Befehlszeilenschritte in diesem Thema in einer Bash-Shell ausführen. Wenn Sie keine Bash-Shell verwenden, erfordern einige Skriptbefehle wie Zeilenfortsetzungszeichen und die Art und Weise, wie Variablen gesetzt und verwendet werden, eine Anpassung für Ihre Shell. Darüber hinaus können die Zitier- und Escape-Regeln für Ihre Shell unterschiedlich sein. Weitere Informationen finden Sie unter Anführungszeichen und Literale mit Zeichenfolgen AWS CLI im AWS Command Line Interface Benutzerhandbuch für Version 2.

Themen

Voraussetzungen für den Einstieg in PCS AWS

Lesen Sie die folgenden Themen, um Ihre AWS-Konto und Ihre lokale Entwicklungsumgebung für AWS PCS vorzubereiten.

Themen

- Melde dich an für AWS und erstellen Sie einen Administratorbenutzer
- Installieren Sie das AWS CLI für AWS PCS
- Erforderliche IAM-Berechtigungen für AWS PCS

Melde dich an für AWS und erstellen Sie einen Administratorbenutzer

Führen Sie die folgenden Aufgaben aus, um den AWS Parallel Computing Service (AWS PCS) einzurichten.

Topics

- Melden Sie sich an für ein AWS-Konto

Melden Sie sich an für ein AWS-Konto

Um loszulegen AWS, benötigen Sie eine AWS-Konto. Informationen zum Erstellen eines AWS-Konto finden Sie unter Erste Schritte mit einem AWS-Konto im AWS -Kontenverwaltung Referenzhandbuch.

Installieren Sie das AWS CLI für AWS PCS

Sie müssen die neueste Version von verwenden AWS CLI. Weitere Informationen finden Sie unter Installation oder Aktualisierung auf die neueste Version von AWS CLI im AWS Command Line Interface Benutzerhandbuch für Version 2.

Sie müssen das konfigurieren AWS CLI. Weitere Informationen finden Sie unter Configure the AWS CLI im AWS Command Line Interface Benutzerhandbuch für Version 2.

Geben Sie an der Befehlszeile den folgenden Befehl ein, um Ihre AWS CLI Daten zu überprüfen. Es sollten Hilfeinformationen angezeigt werden.

```
aws pcs help
```

Erforderliche IAM-Berechtigungen für AWS PCS

Der von Ihnen verwendete IAM-Sicherheitsprinzipal muss über Berechtigungen für die Arbeit mit AWS PCS-IAM-Rollen, serviceverknüpften Rollen AWS CloudFormation, einer VPC und verwandten Ressourcen verfügen. Weitere Informationen finden Sie unter Identity and Access Management für AWS Dienst für parallele Datenverarbeitung und Erstellen einer serviceverknüpften Rolle im Benutzerhandbuch.AWS Identity and Access Management Sie müssen alle Schritte in diesem Handbuch als derselbe Benutzer ausführen. Führen Sie den folgenden Befehl aus, um den aktuellen Benutzer zu überprüfen:

```
aws sts get-caller-identity
```

Verwendung AWS CloudFormation mit dem AWS PCS-Tutorial

Das AWS PCS-Tutorial besteht aus vielen Schritten und soll Ihnen helfen, die Bestandteile eines AWS PCS-Clusters und die zu seiner Erstellung erforderlichen Verfahren zu verstehen. Wir empfehlen, dass Sie die Schritte des Tutorials mindestens einmal durchführen. Sobald Sie ein gutes Verständnis dafür haben, AWS CloudFormation worum es geht, können Sie den Beispielcluster mithilfe von Automatisierung schnell erstellen.

CloudFormation ist ein AWS Service, mit dem Sie AWS Infrastrukturbereitstellungen vorhersehbar und wiederholt erstellen und bereitstellen können. Sie können eine CloudFormation Vorlage verwenden, um die AWS Ressourcen für den Beispielcluster automatisch als einzelne Einheit, einen sogenannten Stack, bereitzustellen. Sie können den Stapel löschen, wenn Sie damit fertig sind.

Weitere Informationen finden Sie unter Erste Schritte mit CloudFormation AWS PCS.

Erstellen Sie eine VPC und Subnetze für PCS AWS

Sie können eine VPC und Subnetze mit einer CloudFormation Vorlage erstellen. Verwenden Sie die folgende URL, um die CloudFormation Vorlage herunterzuladen, und laden Sie sie dann in die CloudFormation Konsole hoch, um einen neuen CloudFormation Stack zu erstellen. Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch unter Verwenden der CloudFormation Konsole.

```
https://aws-hpc-recipes.s3.amazonaws.com/main/recipes/net/hpc_large_scale/assets/
main.yaml
```

Geben Sie bei geöffneter Vorlage in der CloudFormation Konsole die folgenden Optionen ein. Sie können die in der Vorlage bereitgestellten Standardwerte verwenden.

- Gehen Sie unter Geben Sie einen Stacknamen ein:
  - Geben Sie unter Stackname Folgendes ein:

```
hpc-networking
```

- Unter Parameter:
  - Unter VPC:
    - Geben Sie unter CidrBlockFolgendes ein:

```
10.3.0.0/16
```

  - Unter Subnetze A:
    - Geben Sie unter CidrPublicSubnetA Folgendes ein:

```
10.3.0.0/20
```

    - Geben Sie unter CidrPrivateSubnetA Folgendes ein:

```
10.3.128.0/20
```

  - Unter Subnetze B:
    - Geben Sie unter CidrPublicSubnetB Folgendes ein:

```
10.3.16.0/20
```

    - Geben Sie unter CidrPrivateSubnetB Folgendes ein:

```
10.3.144.0/20
```

  - Unter Subnetze C:
    - Wählen Sie für ProvisionSubnetsC die Option True
    - Geben Sie unter CidrPublicSubnetC Folgendes ein:

```
10.3.32.0/20
```

    - Geben Sie unter CidrPrivateSubnetC Folgendes ein:

```
10.3.160.0/20
```

- Unter Fähigkeiten:
  - Markieren Sie das Kästchen Ich bestätige, dass dadurch IAM-Ressourcen erstellt werden AWS CloudFormation könnten.

Überwachen Sie den Status des CloudFormation Stacks. Wenn es erreicht ist`CREATE_COMPLETE`, suchen Sie die ID für die Standardsicherheitsgruppe in der neuen VPC. Sie verwenden die ID später im Tutorial.

Suchen Sie die Standardsicherheitsgruppe für die Cluster-VPC

Gehen Sie wie folgt vor, um die ID für die Standardsicherheitsgruppe in der neuen VPC zu finden:

- Navigieren Sie zur Amazon VPC-Konsole.
- Wählen Sie im VPC-Dashboard die Option Nach VPC filtern aus.
  - Wählen Sie die VPC aus, mit `hpc-networking` der der Name beginnt.
  - Wählen Sie unter Sicherheit die Option Sicherheitsgruppen aus.
- Suchen Sie die Sicherheitsgruppen-ID für die angegebene Gruppe`default`. Sie hat die Beschreibung`default VPC security group`. Sie verwenden die ID später, um EC2-Startvorlagen zu konfigurieren.

Sicherheitsgruppen für AWS PCS erstellen

AWS PCS stützt sich auf Sicherheitsgruppen, um den Netzwerkverkehr in und aus einem Cluster und seinen Compute-Knotengruppen zu verwalten. Ausführliche Informationen zu diesem Thema finden Sie unterAnforderungen und Überlegungen zur Sicherheitsgruppe.

In diesem Schritt verwenden Sie eine CloudFormation Vorlage, um zwei Sicherheitsgruppen zu erstellen.

- Eine Cluster-Sicherheitsgruppe, die die Kommunikation zwischen AWS PCS-Controllern, Rechenknoten und Anmeldeknoten ermöglicht.
- Eine SSH-Sicherheitsgruppe für eingehende Nachrichten, die Sie optional zu Ihren Anmeldeknoten hinzufügen können, um den SSH-Zugriff zu unterstützen

Erstellen Sie die Sicherheitsgruppen für PCS AWS

Sie können eine CloudFormation Vorlage verwenden, um die Sicherheitsgruppen zu erstellen. Verwenden Sie die folgende URL, um die CloudFormation Vorlage herunterzuladen, und laden Sie sie dann in die CloudFormation Konsole hoch, um einen neuen CloudFormation Stack zu erstellen. Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch unter Verwenden der CloudFormation Konsole.

```
https://aws-hpc-recipes.s3.amazonaws.com/main/recipes/pcs/getting_started/assets/pcs-cluster-sg.yaml
```

Geben Sie bei geöffneter Vorlage in der AWS CloudFormation Konsole die folgenden Optionen ein. Beachten Sie, dass einige Optionen in der Vorlage bereits ausgefüllt sind. Sie können sie einfach als Standardwerte beibehalten.

- Unter Geben Sie einen Stacknamen an
  - Geben Sie unter Stackname Folgendes ein:

    ```
    getstarted-sg
    ```
- Unter Parameter
  - Wählen Sie VpcIdunter die VPC aus, mit `hpc-networking` der der Name beginnt.
  - (Optional) Geben Sie unter ClientIpCidreinen restriktiveren IP-Bereich für die eingehende SSH-Sicherheitsgruppe ein. Wir empfehlen, dass Sie dies mit Ihrer eigenen IP/Ihrem eigenen Subnetz einschränken (x.x.x.x/32 für Ihre eigene IP oder x.x.x.x/24 für den Bereich). Ersetzen Sie x.x.x.x durch Ihre eigene ÖFFENTLICHE IP. Sie können Ihre öffentliche IP mithilfe von Tools wie https://ifconfig.co/ abrufen.

Überwachen Sie den Status des CloudFormation Stacks. Wenn es `CREATE_COMPLETE` die Sicherheitsgruppe erreicht, sind die Ressourcen bereit.

Es wurden zwei Sicherheitsgruppen mit den folgenden Namen erstellt:

- `cluster-getstarted-sg`— das ist die Cluster-Sicherheitsgruppe
- `inbound-ssh-getstarted-sg`— Dies ist eine Sicherheitsgruppe, die eingehenden SSH-Zugriff ermöglicht

Erstellen Sie einen Cluster in AWS PCS

In AWS PCS ist ein Cluster eine persistente Ressource für die Verwaltung von Ressourcen und die Ausführung von Workloads. Sie erstellen einen Cluster für einen bestimmten Scheduler (AWS PCS unterstützt derzeit Slurm) in einem Subnetz einer neuen oder vorhandenen VPC. Der Cluster akzeptiert und plant Jobs und startet auch die Rechenknoten (EC2-Instances), die diese Jobs verarbeiten.

Um Ihren Cluster zu erstellen

1. Öffnen Sie die AWS PCS-Konsole und wählen Sie Create Cluster aus.
2. Geben Sie im Abschnitt Clusterdetails die folgenden Felder ein:
   - Clustername — Geben Sie ein `get-started`
   - Scheduler — Wählen Sie Slurm Version 25.11
   - Controller-Größe — Wählen Sie Klein
3. Wählen Sie im Bereich Netzwerk Werte für die folgenden Felder aus:
   - VPC — Wählen Sie die benannte VPC `hpc-networking:Large-Scale-HPC`
   - Subnetz — Wählen Sie das Subnetz aus, mit dem der Name beginnt `hpc-networking:PrivateSubnetA`
   - Sicherheitsgruppen — Wählen Sie die Cluster-Sicherheitsgruppe mit dem Namen `cluster-getstarted-sg`
4. Wählen Sie Cluster erstellen.

> **Note**
>
> Im Feld Status wird während der Bereitstellung des Clusters die Meldung Wird erstellt angezeigt. Die Clustererstellung kann mehrere Minuten dauern.

Erstellen Sie gemeinsam genutzten Speicher für AWS PCS in Amazon Elastic File System

Amazon Elastic File System (Amazon EFS) ist ein AWS Service, der serverlosen, vollständig elastischen Dateispeicher bereitstellt, sodass Sie Dateidaten gemeinsam nutzen können, ohne

Speicherkapazität und Leistung bereitstellen oder verwalten zu müssen. Weitere Informationen finden Sie unter Was ist Amazon Elastic File System? im Amazon Elastic File System-Benutzerhandbuch.

Der AWS PCS-Demonstrationscluster verwendet ein EFS-Dateisystem, um ein gemeinsames Basisverzeichnis zwischen den Clusterknoten bereitzustellen. Erstellen Sie ein EFS-Dateisystem in derselben VPC wie Ihr Cluster.

So erstellen Sie ein Amazon-EFS-Dateisystem

1. Gehen Sie zur Amazon EFS-Konsole.
2. Stellen Sie sicher, dass sie auf die gleiche Einstellung eingestellt ist AWS-Region , auf der Sie AWS PCS ausprobieren möchten.
3. Wählen Sie Create file system (Dateisystem erstellen) aus.
4. Stellen Sie auf der Seite Dateisystem erstellen die folgenden Parameter ein:
   - Für Name geben Sie `getstarted-efs` ein.
   - Wählen Sie unter Virtual Private Cloud (VPC) die VPC mit dem Namen `hpc-networking:Large-Scale-HPC`
   - Wählen Sie Create (Erstellen) aus. Dadurch kehren Sie zur Seite Dateisysteme zurück.
5. Notieren Sie sich die Dateisystem-ID für das `getstarted-efs` Dateisystem. Sie benötigen diese Informationen später.

Erstellen Sie gemeinsamen Speicher für AWS PCS in Amazon FSx for Lustre

Amazon FSx for Lustre macht es einfach und kostengünstig, das beliebte, leistungsstarke Lustre-Dateisystem zu starten und auszuführen. Sie verwenden Lustre für Workloads, bei denen es auf Geschwindigkeit ankommt, wie z. B. maschinelles Lernen, High Performance Computing (HPC), Videoverarbeitung und Finanzmodellierung. Weitere Informationen finden Sie unter Was ist Amazon FSx for Lustre? im Amazon FSx for Lustre-Benutzerhandbuch.

Der AWS PCS-Demonstrationscluster kann ein FSx for Lustre-Dateisystem verwenden, um ein leistungsstarkes gemeinsames Verzeichnis zwischen den Clusterknoten bereitzustellen. Erstellen Sie ein FSx for Lustre-Dateisystem in derselben VPC wie Ihr Cluster.

Um Ihr FSx for Lustre-Dateisystem zu erstellen

1. Gehen Sie zur FSx Amazon-Konsole.
2. Stellen Sie sicher, dass die Konsole so eingestellt ist, dass AWS-Region sie dasselbe verwendet wie Ihr Cluster.
3. Wählen Sie Create file system (Dateisystem erstellen) aus.
   - Wählen Sie unter Dateisystemtyp auswählen die Option Amazon FSx for Lustre und dann Weiter.
4. Stellen Sie auf der Seite „Dateisystemdetails angeben“ die folgenden Parameter ein:
   - Unter Dateisystemdetails
     - Für Name geben Sie `getstarted-fsx` ein.
     - Wählen Sie für Bereitstellung und Speichertyp die Optionen Persistent, SSD
     - Wählen Sie für Durchsatz pro Speichereinheit 125 MB/s/TiB
     - Geben Sie für Speicherkapazität 1,2 TiB ein
     - Wählen Sie für die Metadatenkonfiguration die Option Automatisch
     - Wählen Sie als Datenkomprimierungstyp LZ4
   - Unter Netzwerk und Sicherheit
     - Wählen Sie für Virtual Private Cloud (VPC) die VPC mit dem Namen `hpc-networking:Large-Scale-HPC`
     - Belassen Sie für VPC-Sicherheitsgruppen die Sicherheitsgruppe mit dem Namen `default`
     - Wählen Sie für Subnetz das Subnetz aus, mit dem der Name beginnt `hpc-networking:PrivateSubnetA`
   - Behalten Sie für die anderen Optionen ihre Standardwerte bei.
   - Wählen Sie Weiter.
5. Wählen Sie auf der Seite Überprüfen und erstellen die Option Dateisystem erstellen aus. Dadurch kehren Sie zur Seite Dateisysteme zurück.
6. Navigieren Sie zur Detailseite für das FSx for Lustre-Dateisystem, das Sie erstellt haben.
7. Notieren Sie sich die Dateisystem-ID und den Mount-Namen. Sie benötigen diese Informationen später.

> **Note**
>
> Das Feld Status zeigt Creating an, während das Dateisystem bereitgestellt wird. Die Erstellung des Dateisystems kann mehrere Minuten dauern. Warten Sie, bis der Vorgang abgeschlossen ist, bevor Sie mit dem Rest des Tutorials fortfahren.

Erstellen Sie Compute-Knotengruppen in AWS PCS

Eine Rechenknotengruppe ist eine virtuelle Sammlung von Rechenknoten (EC2-Instances), die AWS PCS startet und verwaltet. Wenn Sie eine Compute-Knotengruppe definieren, geben Sie allgemeine Merkmale wie EC2-Instance-Typen, minimale und maximale Instance-Anzahl, Ziel-VPC-Subnetze, bevorzugte Kaufoption und benutzerdefinierte Startkonfiguration an. AWS PCS startet, verwaltet und beendet Rechenknoten in einer Compute-Knotengruppe gemäß diesen Einstellungen auf effiziente Weise. Der Demonstrationscluster verwendet eine Rechenknotengruppe, um Anmeldeknoten für den Benutzerzugriff bereitzustellen, und eine separate Rechenknotengruppe, um Jobs zu verarbeiten. In den folgenden Themen werden die Verfahren zum Einrichten dieser Compute-Knotengruppen in Ihrem Cluster beschrieben.

Themen

Erstellen Sie ein Instanzprofil für AWS PCS

Compute-Knotengruppen benötigen ein Instanzprofil, wenn sie erstellt werden. Wenn Sie die AWS-Managementkonsole verwenden, um eine Rolle für Amazon EC2 zu erstellen, erstellt die Konsole automatisch ein Instanceprofil und gibt ihm denselben Namen wie der Rolle. Weitere Informationen finden Sie unter Verwenden von Instanzprofilen im AWS Identity and Access Management Benutzerhandbuch.

Im folgenden Verfahren verwenden Sie die, AWS-Managementkonsole um eine Rolle für Amazon EC2 zu erstellen, wodurch auch das Instance-Profil für Ihre Rechenknotengruppen erstellt wird.

Um die Rolle und das Instance-Profil zu erstellen

- Navigieren Sie zur IAM-Konsole.
- Wählen Sie unter Access management (Zugriffsverwaltung) Policies (Richtlinien) aus.
  - Wählen Sie Richtlinie erstellen aus.
  - Wählen Sie unter Berechtigungen angeben für den Richtlinieneditor die Option JSON aus.
  - Ersetzen Sie den Inhalt des Texteditors durch Folgendes:

    JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "pcs:RegisterComputeNodeGroupInstance"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

  - Wählen Sie Weiter aus.
  - Geben Sie unter Überprüfen und erstellen als Richtlinienname den Wert ein`AWSPCS-getstarted-policy`.
  - Wählen Sie Richtlinie erstellen aus.
- Wählen Sie unter Access management (Zugriffsverwaltung) Roles (Rollen) aus.
- Wählen Sie Rolle erstellen aus.
- Unter Vertrauenswürdige Entität auswählen:
  - Wählen Sie für Vertrauenswürdigen Entitätstyp die Option AWS Dienst aus
  - Wählen Sie unter Anwendungsfall die Option EC2 aus.
    - Wählen Sie dann unter Wählen Sie einen Anwendungsfall für den angegebenen Dienst die Option EC2 aus.
  - Wählen Sie Weiter aus.
- Unter Berechtigungen hinzufügen:
  - Suchen Sie unter Permissions policies nach AWSPCS-getstarted-policy.

- Markieren Sie das Kästchen neben AWSPCS-getstarted-policy, um es der Rolle hinzuzufügen.
- Suchen Sie unter Permissions policies nach Amazon SSMManaged InstanceCore.
- Markieren Sie das Kästchen neben Amazon SSMManaged InstanceCore, um es der Rolle hinzuzufügen.
- Wählen Sie Weiter aus.

- Unter Name überprüfen und erstellen:
  - Unter Rollendetails:
    - Geben Sie für Role name (Rollenname) den Namen `AWSPCS-getstarted-role` ein.
  - Wählen Sie Rolle erstellen aus.

Startvorlagen für AWS PCS erstellen

Wenn Sie eine Compute-Knotengruppe erstellen, stellen Sie eine EC2-Startvorlage bereit, die AWS PCS zur Konfiguration der von PCS gestarteten EC2-Instances verwendet. Dazu gehören Einstellungen wie Sicherheitsgruppen und Skripts, die beim Start der Instance ausgeführt werden.

In diesem Schritt wird eine CloudFormation Vorlage verwendet, um zwei EC2-Startvorlagen zu erstellen. Eine Vorlage wird zur Erstellung von Login-Knoten und die andere zur Erstellung von Rechenknoten verwendet. Der Hauptunterschied zwischen ihnen besteht darin, dass die Anmeldeknoten so konfiguriert werden können, dass sie eingehenden SSH-Zugriff ermöglichen.

Greifen Sie auf die Vorlage zu CloudFormation

Verwenden Sie die folgende URL, um die CloudFormation Vorlage herunterzuladen, und laden Sie sie dann in die CloudFormation Konsole hoch, um einen neuen CloudFormation Stack zu erstellen. Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch unter Verwenden der CloudFormation Konsole.

```
https://aws-hpc-recipes.s3.amazonaws.com/main/recipes/pcs/getting_started/assets/pcs-lt-efs-fsxl.yaml
```

Verwenden Sie die CloudFormation Vorlage, um EC2-Startvorlagen zu erstellen

Gehen Sie wie folgt vor, um die CloudFormation Vorlage in der CloudFormation Konsole auszufüllen

- Gehen Sie unter Geben Sie einen Stacknamen ein:

  - Geben Sie unter Stackname den Wert ein`getstarted-lt`.
- Unter Parameter:
  - Unter Sicherheit
    - Wählen Sie für die Sicherheitsgruppe aus VpcSecurityGroupId, die `default` in Ihrer Cluster-VPC benannt ist.
    - Wählen Sie für ClusterSecurityGroupIddie Gruppe mit dem Namen `cluster-getstarted-sg`
    - Wählen Sie für SshSecurityGroupIddie benannte Gruppe aus `inbound-ssh-getstarted-sg`
    - Wählen Sie für SshKeyNameIhr bevorzugtes SSH-Schlüsselpaar aus.
  - Unter Dateisysteme
    - Geben Sie für EfsFilesystemIddie Dateisystem-ID aus dem EFS-Dateisystem ein, das Sie zuvor im Tutorial erstellt haben.
    - Geben Sie für FSxLustreFilesystemIddie Dateisystem-ID aus dem FSx for Lustre-Dateisystem ein, das Sie zuvor im Tutorial erstellt haben.
    - Geben Sie für FSxLustreFilesystemMountNameden Mount-Namen für dasselbe FSx für Lustre-Dateisystem ein.
- Wählen Sie Weiter und dann erneut Weiter.
- Wählen Sie Absenden aus.

Überwachen Sie den Status des CloudFormation Stacks. Wenn `CREATE_COMPLETE` die Startvorlage erreicht ist, kann sie verwendet werden.

> **Note**
>
> Um alle Ressourcen zu sehen, die die CloudFormation Vorlage erstellt hat, öffnen Sie die CloudFormation Konsole. Wählen Sie das `getstarted-lt`-Stack, und wählen Sie dann die Registerkarte Ressourcen.

Erstellen Sie eine Rechenknotengruppe für Anmeldeknoten in AWS PCS

Eine Rechenknotengruppe ist eine virtuelle Sammlung von Rechenknoten (EC2-Instances), die AWS PCS startet und verwaltet. Wenn Sie eine Compute-Knotengruppe definieren, geben Sie allgemeine

Merkmale wie EC2-Instance-Typen, minimale und maximale Instance-Anzahl, Ziel-VPC-Subnetze, bevorzugte Kaufoption und benutzerdefinierte Startkonfiguration an. AWS PCS startet, verwaltet und beendet Rechenknoten in einer Compute-Knotengruppe gemäß diesen Einstellungen auf effiziente Weise.

In diesem Schritt starten Sie eine statische Rechenknotengruppe, die interaktiven Zugriff auf den Cluster bietet. Sie können sich mit SSH oder Amazon EC2 Systems Manager (SSM) anmelden, dann Shell-Befehle ausführen und Slurm-Jobs verwalten.

Um die Compute-Knotengruppe zu erstellen

- Öffnen Sie die AWS PCS-Konsole und navigieren Sie zu Clusters.
- Wählen Sie den Cluster mit dem Namen `get-started`
- Navigieren Sie zu Compute Node Groups und wählen Sie Create aus.
- Geben Sie im Abschnitt Konfiguration der Compute-Knotengruppe Folgendes ein:
  - Name der Knotengruppe berechnen — Geben Sie ein`login`.
- Geben Sie unter Computerkonfiguration die folgenden Werte ein, oder wählen Sie sie aus:
  - EC2-Startvorlage — Wählen Sie die Startvorlage aus, deren Name steht `login-getstarted-lt`
  - IAM-Instanzprofil — Wählen Sie das angegebene Instance-Profil `AWSPCS-getstarted-role`
  - Subnetze — Wählen Sie das Subnetz aus, mit dem der Name beginnt. `hpc-networking:PublicSubnetA`
  - Instanzen — Wählen Sie aus. `c6i.xlarge`
  - Skalierungskonfiguration — Geben Sie 1 für Mindest. Anzahl der Instanzen den Wert ein. Geben 1 Sie für Max. Anzahl der Instanzen den Wert ein.
- Geben Sie unter Zusätzliche Einstellungen Folgendes an:
  - AMI-ID — Wählen Sie ein AMI aus, das Sie verwenden möchten und das einen Namen im folgenden Format hat:

    ```
    aws-pcs-sample_ami-al2023-platform-slurm-version
    ```

    > **Note**
    >
    > Beispiel-AMIs für Slurm 25.05 und frühere Versionen verwenden Amazon Linux 2 (`amzn2`) anstelle von Amazon Linux 2023 (). `al2023`

Weitere Informationen zu den Beispiel-AMIs finden Sie unter. Verwenden von Amazon Machine Images (AMIs) -Beispieldateien mit AWS STK.

- Wählen Sie Create Compute Node Group aus.

Das Feld Status zeigt Creating an, während die Compute-Knotengruppe bereitgestellt wird. Sie können mit dem nächsten Schritt des Tutorials fortfahren, während es in Bearbeitung ist.

Erstellen Sie eine Rechenknotengruppe für die Ausführung von Rechenjobs in AWS PCS

In diesem Schritt starten Sie eine Compute-Knotengruppe, die sich elastisch skalieren lässt, um an den Cluster übermittelte Jobs auszuführen.

Um die Compute-Knotengruppe zu erstellen

- Öffnen Sie die AWS PCS-Konsole und navigieren Sie zu Clusters.
- Wählen Sie den Cluster mit dem Namen `get-started`
- Navigieren Sie zu Compute Node Groups und wählen Sie Create aus.
- Geben Sie im Abschnitt Konfiguration der Compute-Knotengruppe Folgendes ein:
  - Name der Knotengruppe berechnen — Geben Sie ein`compute-1`.
- Geben Sie unter Computerkonfiguration die folgenden Werte ein, oder wählen Sie sie aus:
  - EC2-Startvorlage — Wählen Sie die Startvorlage aus, deren Name steht `compute-getstarted-lt`
  - IAM-Instanzprofil — Wählen Sie das angegebene Instance-Profil `AWSPCS-getstarted-role`
  - Subnetze — Wählen Sie das Subnetz aus, mit dem der Name beginnt. `hpc-networking:PrivateSubnetA`
  - Instanzen — Wählen Sie aus. `c6i.xlarge`
  - Skalierungskonfiguration — Geben Sie `0` für Mindest. Anzahl der Instanzen den Wert ein. Geben `4` Sie für Max. Anzahl der Instanzen den Wert ein.
- Geben Sie unter Zusätzliche Einstellungen Folgendes an:
  - AMI-ID — Wählen Sie ein AMI aus, das Sie verwenden möchten und das einen Namen im folgenden Format hat:

```
aws-pcs-sample_ami-al2023-platform-slurm-version
```

> **Note**
>
> Beispiel-AMIs für Slurm 25.05 und frühere Versionen verwenden Amazon Linux 2 (`amzn2`) anstelle von Amazon Linux 2023 (). `al2023`

Weitere Informationen zu den Beispiel-AMIs finden Sie unter. Verwenden von Amazon Machine Images (AMIs) -Beispieldateien mit AWS STK.

- Wählen Sie Create Compute Node Group aus.

Das Feld Status zeigt Creating an, während die Compute-Knotengruppe bereitgestellt wird.

> **Important**
>
> Warten Sie, bis im Statusfeld Aktiv angezeigt wird, bevor Sie mit dem nächsten Schritt in diesem Tutorial fortfahren.

Erstellen Sie eine Warteschlange zur Verwaltung von Jobs in AWS PCS

Sie reichen einen Job an eine Warteschlange weiter, um ihn auszuführen. Der Job verbleibt in der Warteschlange, bis AWS PCS die Ausführung auf einer Rechenknotengruppe plant. Jede Warteschlange ist einer oder mehreren Rechenknotengruppen zugeordnet, die die für die Verarbeitung erforderlichen EC2 Instanzen bereitstellen.

In diesem Schritt erstellen Sie eine Warteschlange, die die Rechenknotengruppe zur Verarbeitung von Jobs verwendet.

So erstellen Sie eine Warteschlange

- Öffnen Sie die AWS PCS-Konsole.
- Wählen Sie den genannten Cluster aus`get-started`.

- Navigieren Sie zu Compute Node Groups und stellen Sie sicher, dass der Status der `compute-1` Gruppe Aktiv lautet.

> **Important**
>
> Der Status der `compute-1` Gruppe muss Aktiv sein, bevor Sie mit dem nächsten Schritt fortfahren können.

- Navigieren Sie zu Warteschlangen und wählen Sie Warteschlange erstellen.
  - Geben Sie im Abschnitt Warteschlangenkonfiguration die folgenden Werte an:
    - Name der Warteschlange — Geben Sie Folgendes ein: `demo`
    - Compute-Knotengruppen — Wählen Sie die benannte Compute-Knotengruppe aus`compute-1`.
- Wählen Sie Create queue (Warteschlange erstellen) aus.

Während die Warteschlange erstellt wird, wird im Statusfeld Creating angezeigt.

> **Important**
>
> Warten Sie, bis im Statusfeld Aktiv angezeigt wird, bevor Sie mit dem nächsten Schritt in diesem Tutorial fortfahren.

Connect zu Ihrem AWS PCS-Cluster her

Wenn der Status der `login` Compute-Knotengruppe Aktiv lautet, können Sie eine Verbindung zu der von ihr erstellten EC2 Instanz herstellen.

Um eine Verbindung zum Login-Knoten herzustellen

- Öffnen Sie die AWS PCS-Konsole und navigieren Sie zu Clusters.
- Wählen Sie den genannten Cluster aus`get-started`.
- Wählen Sie Compute Node Groups aus.
- Navigieren Sie zu der genannten Compute-Knotengruppe`login`.
- Suchen Sie die Compute-Knotengruppen-ID.

- Öffnen Sie in einem anderen Browserfenster oder einer anderen Registerkarte die EC2 Amazon-Konsole.
  - Wählen Sie Instances.
  - Suchen Sie nach EC2 Instances mit dem folgenden Tag. *node-group-id*Ersetzen Sie ihn durch den Wert der Compute-Knotengruppen-ID aus dem vorherigen Schritt. Es sollte 1 Instanz geben.

    ```
    aws:pcs:compute-node-group-id=node-group-id
    ```

  - Connect zur EC2 Instanz her. Sie können Session Manager oder SSH verwenden.

    Session Manager

    - Wählen Sie die Instance aus.
    - Wählen Sie Connect aus.
    - Wählen Sie unter Mit Instanz verbinden die Option Session Manager aus.
    - Wählen Sie Connect aus.
    - Wählen Sie Connect aus. In Ihrem Browser wird ein interaktives Terminal gestartet.

    SSH

    - Wählen Sie die Instance aus.
    - Wählen Sie Connect aus.
    - Wählen Sie unter Mit Instanz verbinden die Option SSH-Client aus.
    - Folgen Sie den Anweisungen der Konsole.

      > **Note**
      >
      > Der Benutzername für die Instanz ist **`ec2-user`**nicht`root`.

Erkunden Sie die Cluster-Umgebung in AWS PCS

Nachdem Sie sich beim Cluster angemeldet haben, können Sie Shell-Befehle ausführen. Sie können beispielsweise Benutzer wechseln, mit Daten auf gemeinsam genutzten Dateisystemen arbeiten und mit Slurm interagieren.

Benutzer ändern

Wenn Sie sich mit Session Manager beim Cluster angemeldet haben, sind Sie möglicherweise verbunden als`ssm-user`. Dies ist ein spezieller Benutzer, der für Session Manager erstellt wurde. Wechseln Sie mit dem folgenden Befehl zum Standardbenutzer auf Amazon Linux 2023. Sie müssen dies nicht tun, wenn Sie eine Verbindung über SSH hergestellt haben.

```
sudo su - ec2-user
```

Arbeiten Sie mit gemeinsam genutzten Dateisystemen

Mit dem Befehl können Sie überprüfen, ob das EFS-Dateisystem und die Dateisysteme FSx for Lustre verfügbar sind. `df -h` Die Ausgabe auf Ihrem Cluster sollte wie folgt aussehen:

```
[ec2-user@ip-10-3-6-103 ~]$ df -h
Filesystem                 Size  Used Avail Use% Mounted on
devtmpfs                   3.8G     0  3.8G   0% /dev
tmpfs                      3.9G     0  3.9G   0% /dev/shm
tmpfs                      3.9G  556K  3.9G   1% /run
tmpfs                      3.9G     0  3.9G   0% /sys/fs/cgroup
/dev/nvme0n1p1              24G   18G  6.6G  73% /
127.0.0.1:/                8.0E     0  8.0E   0% /home
10.3.132.79@tcp:/zlshxbev  1.2T  7.5M  1.2T   1% /shared
tmpfs                      780M     0  780M   0% /run/user/0
tmpfs                      780M     0  780M   0% /run/user/1000
```

Das `/home` Dateisystem mountet 127.0.0.1 und hat eine sehr große Kapazität. Dies ist das EFS-Dateisystem, das Sie zu Beginn des Tutorials erstellt haben. Alle hier geschriebenen Dateien sind `/home` auf allen Knoten im Cluster unter verfügbar.

Das `/shared` Dateisystem mountet eine private IP und hat eine Kapazität von 1,2 TB. Dies ist das FSx for Lustre-Dateisystem, das Sie zu Beginn des Tutorials erstellt haben. Alle hier geschriebenen Dateien sind `/shared` auf allen Knoten im Cluster unter verfügbar.

Interagiere mit Slurm

Topics

- Listet Warteschlangen und Knoten auf

- Jobs anzeigen

Listet Warteschlangen und Knoten auf

Sie können die Warteschlangen und die Knoten, mit denen sie verknüpft sind, auflisten. `sinfo` Die Ausgabe Ihres Clusters sollte wie folgt aussehen:

```
[ec2-user@ip-10-3-6-103 ~]$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
demo         up   infinite      4  idle~ compute-1-[1-4]
[ec2-user@ip-10-3-6-103 ~]$
```

Notieren Sie sich die benannte Partition`demo`. Ihr Status ist `up` und sie hat maximal 4 Knoten. Es ist Knoten in der `compute-1` Knotengruppe zugeordnet. Wenn Sie die Compute-Knotengruppe bearbeiten und die maximale Anzahl von Instanzen auf 8 erhöhen, würde die Anzahl der Knoten lesen 8 und die Knotenliste würde lesen`compute-1-[1-8]`. Wenn Sie eine zweite Rechenknotengruppe `test` mit dem Namen 4 Knoten erstellen und sie der `demo` Warteschlange hinzufügen würden, würden diese Knoten auch in der Knotenliste angezeigt.

Jobs anzeigen

Sie können alle Jobs in jedem Status auf dem System mit auflisten`squeue`. Die Ausgabe Ihres Clusters sollte wie folgt aussehen:

```
[ec2-user@ip-10-3-6-103 ~]$ squeue
JOBID PARTITION NAME USER ST TIME NODES NODELIST(REASON)
```

Versuchen Sie es später `squeue` erneut, wenn ein Slurm-Job aussteht oder läuft.

Führen Sie einen Einzelknotenjob in AWS PCS aus

Um einen Job mit Slurm auszuführen, bereiten Sie ein Einreichungsskript vor, in dem die Jobanforderungen angegeben sind, und senden es mit dem `sbatch` Befehl an eine Warteschlange. In der Regel erfolgt dies von einem gemeinsam genutzten Verzeichnis aus, sodass die Anmelde- und Rechenknoten über einen gemeinsamen Bereich für den Zugriff auf Dateien verfügen.

Connect zum Login-Knoten Ihres Clusters her und führen Sie die folgenden Befehle an der Shell-Eingabeaufforderung aus.

- Werden Sie der Standardbenutzer. Wechseln Sie in das gemeinsam genutzte Verzeichnis.

```
sudo su - ec2-user
cd /shared
```

- Verwenden Sie die folgenden Befehle, um ein Beispiel-Jobskript zu erstellen:

```
cat << EOF > job.sh
#!/bin/bash
#SBATCH -J single
#SBATCH -o single.%j.out
#SBATCH -e single.%j.err

echo "This is job \${SLURM_JOB_NAME} [\${SLURM_JOB_ID}] running on \
${SLURMD_NODENAME}, submitted from \${SLURM_SUBMIT_HOST}" && sleep 60 && echo "Job
 complete"
EOF
```

- Senden Sie das Jobskript an den Slurm-Scheduler:

```
sbatch -p demo job.sh
```

- Wenn der Job eingereicht wird, wird eine Job-ID als Zahl zurückgegeben. Verwenden Sie diese ID, um den Jobstatus zu überprüfen. Ersetzen Sie *`job-id`* den folgenden Befehl durch die Zahl, die von zurückgegeben wurde`sbatch`.

```
squeue --job job-id
```

  Example

```
squeue --job 1
```

  Der `squeue` Befehl gibt eine Ausgabe zurück, die der folgenden ähnelt:

```
JOBID PARTITION NAME USER     ST TIME NODES NODELIST(REASON)
1     demo      test ec2-user CF 0:47 1     compute-1
```

- Überprüfen Sie weiterhin den Status des Jobs, bis er den Status `R` (läuft) erreicht. Der Job ist erledigt, wenn `squeue` nichts zurückgegeben wird.
- Untersuchen Sie den Inhalt des `/shared` Verzeichnisses.

```
ls -alth /shared
```

Die Befehlsausgabe ähnelt der folgenden:

```
-rw-rw-r- 1 ec2-user ec2-user 107 Mar 19 18:33 single.1.out
-rw-rw-r- 1 ec2-user ec2-user 0 Mar 19 18:32 single.1.err
-rw-rw-r- 1 ec2-user ec2-user 381 Mar 19 18:29 job.sh
```

Die Dateien sind benannt `single.1.out` und `single.1.err` wurden von einem der Rechenknoten Ihres Clusters geschrieben. Da der Job in einem gemeinsam genutzten Verzeichnis (`/shared`) ausgeführt wurde, sind sie auch auf Ihrem Anmeldeknoten verfügbar. Aus diesem Grund haben Sie für diesen Cluster ein FSx For Lustre-Dateisystem konfiguriert.

- Untersuchen Sie den Inhalt der `single.1.out` Datei.

```
cat /shared/single.1.out
```

Die Ausgabe sieht folgendermaßen oder ähnlich aus:

```
This is job test [1] running on compute-1, submitted from ip-10-3-13-181
Job complete
```

Führen Sie einen MPI-Job mit mehreren Knoten mit Slurm in PCS aus AWS

Diese Anweisungen demonstrieren die Verwendung von Slurm zur Ausführung eines MPI-Jobs (Message Passing Interface) in PCS. AWS

Führen Sie die folgenden Befehle an einer Shell-Eingabeaufforderung Ihres Login-Knotens aus.

- Werden Sie der Standardbenutzer. Wechseln Sie in sein Home-Verzeichnis.

```
sudo su - ec2-user
cd ~/
```

- Erstellen Sie Quellcode in der Programmiersprache C.

```
cat > hello.c << EOF
// * mpi-hello-world - https://www.mpitutorial.com
// Released under MIT License
//
// Copyright (c) 2014 MPI Tutorial.
//
// 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, subject to the following conditions:
// The above copyright notice and this permission notice shall be included in
// all copies or substantial portions of the Software.
//
// 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.

#include <mpi.h>
#include <stdio.h>
#include <stddef.h>

int main(int argc, char** argv) {
  // Initialize the MPI environment. The two arguments to MPI Init are not
  // currently used by MPI implementations, but are there in case future
  // implementations might need the arguments.
  MPI_Init(NULL, NULL);

  // Get the number of processes
  int world_size;
  MPI_Comm_size(MPI_COMM_WORLD, &world_size);

  // Get the rank of the process
  int world_rank;
  MPI_Comm_rank(MPI_COMM_WORLD, &world_rank);

  // Get the name of the processor
  char processor_name[MPI_MAX_PROCESSOR_NAME];
```

```
  int name_len;
  MPI_Get_processor_name(processor_name, &name_len);

  // Print off a hello world message
  printf("Hello world from processor %s, rank %d out of %d processors\n",
         processor_name, world_rank, world_size);

  // Finalize the MPI environment. No more MPI calls can be made after this
  MPI_Finalize();
}
EOF
```

- Laden Sie das OpenMPI-Modul.

```
module load openmpi
```

- Kompilieren Sie das C-Programm.

```
mpicc -o hello hello.c
```

- Schreiben Sie ein Slurm-Job-Skript.

```
cat > hello.sh << EOF
#!/bin/bash
#SBATCH -J multi
#SBATCH -o multi.out
#SBATCH -e multi.err
#SBATCH --exclusive
#SBATCH --nodes=4
#SBATCH --ntasks-per-node=1

srun $HOME/hello
EOF
```

- Wechseln Sie in das gemeinsam genutzte Verzeichnis.

```
cd /shared
```

- Reichen Sie das Jobskript ein.

```
sbatch -p demo ~/hello.sh
```

- Wird verwendet`squeue`, um den Job zu überwachen, bis er erledigt ist.
- Überprüfen Sie den Inhalt von`multi.out`:

```
cat multi.out
```

Die Ausgabe sieht folgendermaßen oder ähnlich aus. Beachten Sie, dass jeder Rang seine eigene IP-Adresse hat, da er auf einem anderen Knoten lief.

```
Hello world from processor ip-10-3-133-204, rank 0 out of 4 processors
Hello world from processor ip-10-3-128-219, rank 2 out of 4 processors
Hello world from processor ip-10-3-141-26, rank 3 out of 4 processors
Hello world from processor ip-10-3-143-52, rank 1 out of 4 processor
```

Löschen Sie Ihre AWS Ressourcen für AWS PCS

Nachdem Sie mit den Cluster- und Knotengruppen fertig sind, die Sie für dieses Tutorial erstellt haben, sollten Sie die von Ihnen erstellten Ressourcen löschen.

> **Important**
>
> Sie erhalten Abrechnungsgebühren für alle Ressourcen, die in Ihrem AWS-Konto

Um AWS PCS-Ressourcen zu löschen, die Sie für dieses Tutorial erstellt haben

- Öffnen Sie die AWS PCS-Konsole.
- Navigieren Sie zu dem Cluster mit dem Namen get-started.
- Navigieren Sie zum Abschnitt Warteschlangen.
- Wählen Sie die Warteschlange mit dem Namen demo aus.
- Wählen Sie Löschen.

> **Important**
>
> Warten Sie, bis die Warteschlange gelöscht wurde, bevor Sie fortfahren.

- Navigieren Sie zum Abschnitt Knotengruppen berechnen.

- Wählen Sie die Compute-Knotengruppe mit dem Namen compute-1 aus.
- Wählen Sie Löschen.
- Wählen Sie die Compute-Knotengruppe mit dem Namen login aus.
- Wählen Sie Löschen.

> **Important**
>
> Warten Sie, bis beide Compute-Knotengruppen gelöscht wurden, bevor Sie fortfahren.

- Wählen Sie auf der Cluster-Detailseite für Erste Schritte die Option Löschen aus.

> **Important**
>
> Warten Sie, bis der Cluster gelöscht wurde, bevor Sie mit den nächsten Schritten fortfahren.

Um andere AWS Ressourcen zu löschen, die Sie für dieses Tutorial erstellt haben

- Öffnen Sie die IAM-Konsole.
  - Wählen Sie Roles.
  - Wählen Sie die Rolle mit dem Namen AWSPCS-getstarted-role aus und klicken Sie dann auf Löschen.
  - Nachdem die Rolle gelöscht wurde, wählen Sie Richtlinien aus.
  - Wählen Sie die Richtlinie mit dem Namen AWSPCS-getstarted-policy und anschließend Löschen aus.
- Öffnen Sie die CloudFormation -Konsole.
  - Wählen Sie den Stack mit dem Namen getstarted-lt aus.
  - Wählen Sie Löschen.

> **Important**
>
> Warten Sie, bis der Stapel gelöscht ist, bevor Sie fortfahren.

- Öffnen Sie die Amazon-ECS-Konsole.
  - Wählen Sie Dateisysteme aus.

- Wählen Sie das Dateisystem mit dem Namen getstarted-efs aus.
- Wählen Sie Löschen.

> **Important**
>
> Warten Sie, bis das Dateisystem gelöscht ist, bevor Sie fortfahren.

- Öffnen Sie die FSx Amazon-Konsole.
  - Wählen Sie Dateisysteme aus.
  - Wählen Sie das Dateisystem mit dem Namen getstarted-fsx aus.
  - Wählen Sie Löschen.

> **Important**
>
> Warten Sie, bis das Dateisystem gelöscht ist, bevor Sie fortfahren.

- Öffnen Sie die CloudFormation -Konsole.
  - Wählen Sie den Stack mit dem Namen getstarted-sg aus.
  - Wählen Sie Löschen.
- Öffnen Sie die CloudFormation -Konsole.
  - Wählen Sie den Stack mit dem Namen hpc-networking aus.
  - Wählen Sie Löschen.

Erste Schritte mit CloudFormation AWS PCS

Sie können es verwenden AWS CloudFormation , um einen AWS PCS-Cluster zu erstellen. CloudFormation ermöglicht es Ihnen, AWS Infrastrukturbereitstellungen vorhersehbar und wiederholt zu erstellen und bereitzustellen. Sie können CloudFormation die automatische Bereitstellung von Ressourcen aus vielen AWS Diensten verwenden, um äußerst zuverlässige, skalierbare und kostengünstige Anwendungen zu erstellen, AWS Cloud ohne die zugrunde liegende Infrastruktur erstellen und konfigurieren zu müssen. AWS CloudFormation ermöglicht es Ihnen, mithilfe einer Vorlagendatei eine Sammlung von Ressourcen zu einer einzigen Einheit, einem sogenannten Stapel, zu erstellen und zu löschen. Weitere Informationen zu CloudFormation finden Sie unter Was ist CloudFormation? im AWS CloudFormation Benutzerhandbuch. Weitere Informationen zu AWS PCS-Ressourcentypen finden Sie in CloudFormation der Referenz zu AWS PCS-Ressourcentypen im AWS CloudFormation Benutzerhandbuch.

Themen

Verwenden Sie CloudFormation um ein Beispiel zu erstellen AWS PCS-Cluster

Das folgende Verfahren verwendet eine CloudFormation Vorlage im AWS-Managementkonsole , um einen AWS PCS-Beispielcluster zu erstellen. Weitere Informationen zu CloudFormation finden Sie unter Was ist CloudFormation? im AWS CloudFormation Benutzerhandbuch. Weitere Informationen zu AWS PCS-Ressourcentypen finden Sie in CloudFormation der Referenz zu AWS PCS-Ressourcentypen im AWS CloudFormation Benutzerhandbuch.

Um den Beispielcluster zu erstellen

1. Wählen Sie AWS-Region den aus, in dem der Cluster erstellt werden soll (der Link öffnet die CloudFormation Konsole mit der Vorlage):

- USA Ost (Nord-Virginia) (us-east-1)
- USA Ost (Ohio) (us-east-2)
- USA West (Oregon) (us-west-2)
- Asien-Pazifik (Mumbai) (ap-south-1)
- Asien-Pazifik (Singapur) (ap-southeast-1)
- Asien-Pazifik (Sydney) (ap-southeast-2)
- Asien-Pazifik (Tokio) (ap-northeast-1)
- Asien-Pazifik (Osaka) (ap-northeast-3)
- Europa (Frankfurt) (eu-central-1)
- Europa (Irland) (eu-west-1)
- Europa (London) (eu-west-2)
- Europa (Paris) (eu-west-3)
- Europa (Mailand) (eu-south-1)
- Europa (Spanien) (eu-south-2)
- Europa (Stockholm) (eu-north-1)
- Südamerika (São Paulo) (sa-east-1)
- AWS GovCloud (US-East) (us-gov-east-1)
- AWS GovCloud () (US-Regierung West-1) US-West

2. Geben Sie unter Geben Sie einen Stacknamen an einen beschreibenden Namen ein. Dies ist der Name für Ihren CloudFormation Stack. Die Vorlage verwendet diesen Wert als Namen für Ihren AWS PCS-Cluster.
3. Unter Parameter:

   a. Wählen Sie unter SlurmVersiondie Version von Slurm aus, die Ihr Cluster verwenden soll.

   b. Wählen Sie unter x86 aus NodeArchitecture, um einen Cluster bereitzustellen, der x86_64-kompatible Instances verwendet, oder wählen Sie Graviton, um Arm64-Instanzen zu verwenden.

   c. Wählen Sie für KeyNameein SSH-Schlüsselpaar für den Zugriff auf die Cluster-Anmeldeknoten. Stellen Sie sicher, dass Sie die PEM-Datei für das von Ihnen gewählte key pair haben.

   d. Geben Sie für ClientIpCidreinen IP-Bereich im CIDR-Format ein, um den Zugriff auf die Anmeldeknoten zu steuern.

> **Warning**
>
> Der Standardwert von `0.0.0.0/0` ermöglicht den Zugriff von allen IP-Adressen aus.

e. Behalten Sie die Werte für HpcRecipesS3Bucketund HpcRecipesBranchals Standardwerte bei.

4. Unter Funktionen und Transformationen:

   a. Aktivieren Sie das Kontrollkästchen, um zu bestätigen, dass dadurch IAM-Ressourcen erstellt CloudFormation werden.

   b. Aktivieren Sie das Kontrollkästchen, um zu bestätigen, CloudFormation dass IAM-Ressourcen mit benutzerdefinierten Namen erstellt werden.

   c. Aktivieren Sie das Kontrollkästchen, um den neuen Stack zu bestätigen`CAPABILITY_AUTO_EXPAND`. Weitere Informationen finden Sie unter CreateStack in der AWS CloudFormation -API-Referenz.

5. Wählen Sie Stack erstellen aus.

6. Überwachen Sie den Status Ihres Stacks. Sie können eine Verbindung zum Cluster herstellen, wenn der Status des Stacks lautet`CREATE_COMPLETE`.

Stellen Sie eine Connect zu einem AWS PCS-Cluster her, der erstellt wurde mit CloudFormation

Nachdem Sie einen AWS PCS-Cluster anhand einer CloudFormation Vorlage erstellt haben, können Sie den Cluster mit der AWS PCS-Konsole (im AWS-Managementkonsole) verwalten. Sie können auch eine Verbindung zu einem der Anmeldeknoten des Clusters herstellen, um den Cluster zu verwalten, Jobs auszuführen und Daten zu verwalten. Der CloudFormation Stack bietet Links, über die Sie eine Verbindung zu Ihrem Cluster herstellen können.

Um eine Verbindung zu Ihrem Cluster herzustellen

1. Öffnen Sie die CloudFormation -Konsole.
2. Wählen Sie den Stack aus, den Sie erstellt haben.
3. Wählen Sie die Registerkarte Ausgaben des Stacks.

Der Stapel bietet die folgenden Links:

- PcsConsoleUrl— Wählen Sie diesen Link, um die AWS PCS-Konsole mit dem ausgewählten Cluster zu öffnen. Sie können ihn verwenden, um die Cluster-, Knotengruppen- und Warteschlangenkonfigurationen zu erkunden.
- Ec2 ConsoleUrl — Wählen Sie diesen Link, um die Amazon EC2 EC2-Konsole zu öffnen, die so gefiltert ist, dass die Instances angezeigt werden, die von der Login-Knotengruppe des Clusters verwaltet werden.

  In dieser Ansicht können Sie eine Instanz auswählen und Connect wählen. Die Instanz des Beispielclusters unterstützt eingehendes SSH und AWS Systems Manager Verbindungen in einem Webbrowser. Weitere Informationen finden Sie unter Connect zu Ihrem AWS PCS-Cluster her.

  Nachdem Sie eine Verbindung zu einer Anmeldeinstanz hergestellt haben, können Sie dem Tutorial unter folgen. Erkunden Sie die Cluster-Umgebung in AWS PCS

Bereinigen Sie einen AWS PCS-Cluster in CloudFormation

Wenn Sie früher CloudFormation Ihren AWS PCS-Cluster erstellt haben, können Sie die CloudFormation Konsole öffnen und den Stack löschen, um den Cluster und alle zugehörigen Ressourcen zu löschen.

> **Important**
>
> Wenn Sie für den Beispielcluster zusätzliche Compute-Knotengruppen oder Warteschlangen in Ihrem Cluster erstellt haben (zusätzlich zu den `login` `compute-1` Gruppen, die mit der CloudFormation Beispielvorlage erstellt wurden), müssen Sie die AWS PCS-Konsole verwenden oder AWS CLI diese Ressourcen löschen, bevor Sie den CloudFormation Stack löschen. Weitere Informationen finden Sie unter Löschen eines Clusters in AWS PCS.

Teile einer CloudFormation Vorlage für AWS STK.

Eine CloudFormation Vorlage besteht aus einem oder mehreren Abschnitten, die jeweils einem bestimmten Zweck dienen. CloudFormation definiert das Standardformat, die Syntax und die

Standardsprache in einer Vorlage. Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch unter Arbeiten mit CloudFormation Vorlagen.

CloudFormation Vorlagen sind in hohem Maße anpassbar und daher können ihre Formate variieren. Um zu verstehen, welche Teile einer CloudFormation Vorlage zur Erstellung eines AWS PCS-Clusters erforderlich sind, empfehlen wir Ihnen, sich die Beispielvorlage anzusehen, die wir zur Erstellung eines Beispielclusters zur Verfügung stellen. In diesem Thema werden die Abschnitte dieser Beispielvorlage kurz erläutert.

> **Important**
>
> Die Codebeispiele in diesem Thema sind nicht vollständig. Das Vorhandensein von Auslassungspunkten (`[...]`) weist darauf hin, dass zusätzlicher Code nicht angezeigt wird. Informationen zum Herunterladen der vollständigen YAML-formatted CloudFormation Vorlage finden Sie unterCloudFormation Vorlagen zum Erstellen eines Musters AWS PCS-Cluster.

Inhalt

Header

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Description: AWS Parallel Computing Service "getting started" cluster
```

`AWSTemplateFormatVersion`identifiziert die Version im Vorlagenformat, der die Vorlage entspricht. Weitere Informationen finden Sie unter Versionssyntax für das CloudFormation Vorlagenformat im AWS CloudFormation Benutzerhandbuch.

`Transform`gibt ein Makro an, das zur Verarbeitung der Vorlage CloudFormation verwendet wird. Weitere Informationen finden Sie im Abschnitt Transformieren von CloudFormation

Vorlagen im AWS CloudFormation Benutzerhandbuch. Die `AWS::Serverless-2016-10-31` Transformation ermöglicht CloudFormation die Verarbeitung einer Vorlage, die in der Syntax AWS Serverless Application Model (AWS SAM) geschrieben ist. Weitere Informationen finden Sie unter `AWS::ServerlessTransform` im AWS CloudFormation Benutzerhandbuch.

Metadaten

```
### Stack metadata
Metadata:
  AWS::CloudFormation::Interface:
    ParameterGroups:
      - Label:
          default: PCS Cluster configuration
        Parameters:
          - SlurmVersion
          - ManagedAccounting
          - AccountingPolicyEnforcement
      - Label:
          default: PCS ComputeNodeGroups configuration
        Parameters:
          - NodeArchitecture
          - KeyName
          - ClientIpCidr
      - Label:
          default: HPC Recipes configuration
        Parameters:
          - HpcRecipesS3Bucket
          - HpcRecipesBranch
```

Der `metadata` Abschnitt einer CloudFormation Vorlage enthält Informationen über die Vorlage selbst. Mit der Beispielvorlage wird ein vollständiger HPC-Cluster (High Performance Computing) erstellt, der AWS PCS verwendet. Im Metadatenbereich der Beispielvorlage werden Parameter deklariert, die steuern, wie der entsprechende CloudFormation Stack gestartet (bereitgestellt) wird. Es gibt Parameter, die die Architekturauswahl (`NodeArchitecture`), die Slurm-Version (`SlurmVersion`) und die Zugriffskontrollen (`KeyNameundClientIpCidr`) steuern.

Parameters

`Parameters`In diesem Abschnitt werden die benutzerdefinierten Parameter für die Vorlage definiert. CloudFormation verwendet diese Parameterdefinitionen, um das Formular zu erstellen und zu validieren, mit dem Sie interagieren, wenn Sie einen Stack von dieser Vorlage aus starten.

```
Parameters:

  NodeArchitecture:
    Type: String
    Default: x86
    AllowedValues:
      - x86
      - Graviton
    Description: Processor architecture for the login and compute node instances

  SlurmVersion:
    Type: String
    Default: 25.11
    Description: Version of Slurm to use
    AllowedValues:
         - 25.05
         - 25.11

  ManagedAccounting:
    Type: String
    Default: 'disabled'
    AllowedValues:
      - 'enabled'
      - 'disabled'
    Description: Monitor cluster usage, manage access control, and enforce resource
 limits with Slurm accounting.

  AccountingPolicyEnforcement:
    Description: Specify which Slurm accounting policies to enforce
    Type: String
    Default: none
    AllowedValues:
      - none
      - 'associations,limits,safe'

  KeyName:
    Description: SSH keypair to log in to the head node
    Type: AWS::EC2::KeyPair::KeyName
    AllowedPattern: ".+"  # Required

  ClientIpCidr:
    Description: IP(s) allowed to access the login node over SSH. We recommend that
 you restrict it with your own IP/subnet (x.x.x.x/32 for your own ip or x.x.x.x/24 for
```

```
range. Replace x.x.x.x with your own PUBLIC IP. You can get your public IP using tools
such as https://ifconfig.co/)
   Default: 127.0.0.1/32
   Type: String
   AllowedPattern: (\d{1,3})\.(\d{1,3})\.(\d{1,3})\.(\d{1,3})/(\d{1,2})
   ConstraintDescription: Value must be a valid IP or network range of the form
x.x.x.x/x.

 HpcRecipesS3Bucket:
   Type: String
   Default: aws-hpc-recipes
   Description: HPC Recipes for AWS S3 bucket
   AllowedValues:
        - aws-hpc-recipes
        - aws-hpc-recipes-dev
 HpcRecipesBranch:
   Type: String
   Default: main
   Description: HPC Recipes for AWS release branch
   AllowedPattern: '^(?!.*/\.git$)(?!.*/\.)(?!.*\\.\.)[a-zA-Z0-9-_\.]+$'
```

Mappings

Der `Mappings` Abschnitt definiert Schlüssel-Wert-Paare, die Werte auf der Grundlage bestimmter Bedingungen oder Abhängigkeiten angeben.

```
Mappings:

  Architecture:
    AmiArchParameter:
      Graviton: arm64
      x86: x86_64
    LoginNodeInstances:
      Graviton: c7g.xlarge
      x86: c6i.xlarge
    ComputeNodeInstances:
      Graviton: c7g.xlarge
      x86: c6i.xlarge
```

Ressourcen

`Resources`In diesem Abschnitt werden die AWS Ressourcen, die bereitgestellt und konfiguriert werden sollen, als Teil des Stacks deklariert.

```
Resources:

  [...]
```

Die Vorlage stellt die Beispiel-Cluster-Infrastruktur in Schichten bereit. Es beginnt mit `Networking` der VPC-Konfiguration. Der Speicher wird von dualen Systemen bereitgestellt: `EfsStorage` für gemeinsam genutzten Speicher und `FSxLStorage` für Hochleistungsspeicher. Der Core-Cluster wird durch eingerichtet`PCSCluster`.

```
  Networking:
    Type: AWS::CloudFormation::Stack
    Properties:
      Parameters:
        ProvisionSubnetsC: "False"
      TemplateURL: !Sub 'https://${HpcRecipesS3Bucket}.s3.amazonaws.com/
${HpcRecipesBranch}/recipes/net/hpc_large_scale/assets/main.yaml'

  EfsStorage:
    Type: AWS::CloudFormation::Stack
    Properties:
      Parameters:
        SubnetIds: !GetAtt [ Networking, Outputs.DefaultPrivateSubnet ]
        SubnetCount: 1
        VpcId: !GetAtt [ Networking, Outputs.VPC ]
      TemplateURL: !Sub 'https://${HpcRecipesS3Bucket}.s3.amazonaws.com/
${HpcRecipesBranch}/recipes/storage/efs_simple/assets/main.yaml'

  FSxLStorage:
    Type: AWS::CloudFormation::Stack
    Properties:
      Parameters:
        PerUnitStorageThroughput: 125
        SubnetId: !GetAtt [ Networking, Outputs.DefaultPrivateSubnet ]
        VpcId: !GetAtt [ Networking, Outputs.VPC ]
```

```
      TemplateURL: !Sub 'https://${HpcRecipesS3Bucket}.s3.amazonaws.com/
${HpcRecipesBranch}/recipes/storage/fsx_lustre/assets/persistent.yaml'

  [...]

  # Cluster
  PCSCluster:
    Type: AWS::PCS::Cluster
    Properties:
      Name: !Sub '${AWS::StackName}'
      Size: SMALL
      Scheduler:
        Type: SLURM
        Version: !Ref SlurmVersion
      Networking:
        SubnetIds:
          - !GetAtt [ Networking, Outputs.DefaultPrivateSubnet ]
        SecurityGroupIds:
          - !GetAtt [ PCSSecurityGroup, Outputs.ClusterSecurityGroupId ]
```

Für Rechenressourcen erstellt die Vorlage zwei Knotengruppen: `PCSNodeGroupLogin` für einen einzelnen Anmeldeknoten und `PCSNodeGroupCompute` für bis zu vier Rechenknoten. Diese Knotengruppen werden von `PCSInstanceProfile` für Berechtigungen und beispielsweise `PCSLaunchTemplate` für Konfigurationen unterstützt.

```
  # Compute Node groups
  PCSInstanceProfile:
    Type: AWS::CloudFormation::Stack
    Properties:
      Parameters:
        # We have to regionalize this in case CX use the template in more than one
 region. Otherwise,
        # the create action will fail since instance-role-${AWS::StackName} already
 exists!
        RoleName: !Sub '${AWS::StackName}-${AWS::Region}'
      TemplateURL: !Sub 'https://${HpcRecipesS3Bucket}.s3.amazonaws.com/
${HpcRecipesBranch}/recipes/pcs/getting_started/assets/pcs-iip-minimal.yaml'

  PCSLaunchTemplate:
    Type: AWS::CloudFormation::Stack
    Properties:
      Parameters:
```

```
        VpcDefaultSecurityGroupId: !GetAtt [ Networking, Outputs.SecurityGroup ]
        ClusterSecurityGroupId: !GetAtt [ PCSSecurityGroup,
 Outputs.ClusterSecurityGroupId ]
        SshSecurityGroupId: !GetAtt [ PCSSecurityGroup,
 Outputs.InboundSshSecurityGroupId ]
        EfsFilesystemSecurityGroupId: !GetAtt [ EfsStorage, Outputs.SecurityGroupId ]
        FSxLustreFilesystemSecurityGroupId: !GetAtt [ FSxLStorage,
 Outputs.FSxLustreSecurityGroupId ]
        SshKeyName: !Ref KeyName
        EfsFilesystemId: !GetAtt [ EfsStorage, Outputs.EFSFilesystemId ]
        FSxLustreFilesystemId: !GetAtt [ FSxLStorage, Outputs.FSxLustreFilesystemId ]
        FSxLustreFilesystemMountName: !GetAtt [ FSxLStorage,
 Outputs.FSxLustreMountName ]
      TemplateURL: !Sub 'https://${HpcRecipesS3Bucket}.s3.amazonaws.com/
${HpcRecipesBranch}/recipes/pcs/getting_started/assets/cfn-pcs-lt-efs-fsxl.yaml'

  # Compute Node groups - Login Nodes
  PCSNodeGroupLogin:
    Type: AWS::PCS::ComputeNodeGroup
    Properties:
      ClusterId: !GetAtt [PCSCluster, Id]
      Name: login
      ScalingConfiguration:
        MinInstanceCount: 1
        MaxInstanceCount: 1
      IamInstanceProfileArn: !GetAtt [ PCSInstanceProfile, Outputs.InstanceProfileArn ]
      CustomLaunchTemplate:
        TemplateId: !GetAtt [ PCSLaunchTemplate, Outputs.LoginLaunchTemplateId ]
        Version: 1
      SubnetIds:
        - !GetAtt [ Networking, Outputs.DefaultPublicSubnet ]
      AmiId: !GetAtt [PcsSampleAmi, AmiId]
      InstanceConfigs:
        - InstanceType: !FindInMap [ Architecture, LoginNodeInstances, !Ref
 NodeArchitecture ]

  # Compute Node groups - Compute Nodes
  PCSNodeGroupCompute:
    Type: AWS::PCS::ComputeNodeGroup
    Properties:
      ClusterId: !GetAtt [PCSCluster, Id]
      Name: compute-1
      ScalingConfiguration:
        MinInstanceCount: 0
```

```
      MaxInstanceCount: 4
    IamInstanceProfileArn: !GetAtt [ PCSInstanceProfile, Outputs.InstanceProfileArn ]
    CustomLaunchTemplate:
      TemplateId: !GetAtt [ PCSLaunchTemplate, Outputs.ComputeLaunchTemplateId ]
      Version: 1
    SubnetIds:
      - !GetAtt [ Networking, Outputs.DefaultPrivateSubnet ]
    AmiId: !GetAtt [PcsSampleAmi, AmiId]
    InstanceConfigs:
      - InstanceType: !FindInMap [ Architecture, ComputeNodeInstances, !Ref
NodeArchitecture ]
```

Job Arbeitsplanung erfolgt überPCSQueueCompute.

```
PCSQueueCompute:
  Type: AWS::PCS::Queue
  Properties:
    ClusterId: !GetAtt [PCSCluster, Id]
    Name: demo
    ComputeNodeGroupConfigurations:
      - ComputeNodeGroupId: !GetAtt [PCSNodeGroupCompute, Id]
```

Die AMI-Auswahl erfolgt automatisch über die PcsAMILookupFn Lambda-Funktion und zugehörige Ressourcen.

```
PcsAMILookupRole:
  Type: AWS::IAM::Role
  [...]

PcsAMILookupFn:
  Type: AWS::Lambda::Function
  Properties:
    Runtime: python3.12
    Handler: index.handler
    Role: !GetAtt PcsAMILookupRole.Arn
    Code:
      [...]
    Timeout: 30
    MemorySize: 128
```

```
  # Example of using the custom resource to look up an AMI
  PcsSampleAmi:
    Type: Custom::AMILookup
    Properties:
      ServiceToken: !GetAtt PcsAMILookupFn.Arn
      OperatingSystem: 'amzn2'
      Architecture: !FindInMap [ Architecture, AmiArchParameter, !Ref
 NodeArchitecture ]
      SlurmVersion: !Ref SlurmVersion
```

Outputs

Die Vorlage gibt Cluster-Identifizierungs- und Verwaltungs-URLs über `ClusterIdPcsConsoleUrl`, und `Ec2ConsoleUrl` aus.

```
Outputs:
  ClusterId:
    Description: The Id of the PCS cluster
    Value: !GetAtt [ PCSCluster, Id ]

  PcsConsoleUrl:
    Description: URL to access the cluster in the PCS console
    Value: !Sub
      - https://${ConsoleDomain}/pcs/home?region=${AWS::Region}#/clusters/${ClusterId}
      - { ConsoleDomain: !If [ GovCloud, 'console.amazonaws-us-gov.com', !If [ China,
 'console.amazonaws.cn', !Sub '${AWS::Region}.console.aws.amazon.com']],
          ClusterId: !GetAtt [ PCSCluster, Id ]
        }
    Export:
      Name: !Sub ${AWS::StackName}-PcsConsoleUrl

  Ec2ConsoleUrl:
    Description: URL to access instance(s) in the login node group via Session Manager
    Value: !Sub
      - https://${ConsoleDomain}/ec2/home?region=
${AWS::Region}#Instances:instanceState=running;tag:aws:pcs:compute-node-group-id=
${NodeGroupLoginId}
      - { ConsoleDomain: !If [ GovCloud, 'console.amazonaws-us-gov.com', !If [ China,
 'console.amazonaws.cn', !Sub '${AWS::Region}.console.aws.amazon.com']],
          NodeGroupLoginId: !GetAtt [ PCSNodeGroupLogin, Id ]
        }
    Export:
```

```
      Name: !Sub ${AWS::StackName}-Ec2ConsoleUrl
```

CloudFormation Vorlagen zum Erstellen eines Musters AWS PCS-Cluster

AWS-Region Name	AWS-Region	Quelle ansehen	Stack starten
USA Ost (Nord-Virginia)	us-east-1	Laden Sie YAML herunter	
USA Ost (Ohio)	us-east-2	Laden Sie YAML herunter	
USA West (Oregon)	us-west-2	Laden Sie YAML herunter	
Asien-Pazifik (Mumbai)	ap-south-1	Laden Sie YAML herunter	
Asien-Pazifik (Singapur)	ap-southeast-1	Laden Sie YAML herunter	
Asien-Pazifik (Sydney)	ap-southeast-2	Laden Sie YAML herunter	
Asien-Pazifik (Tokio)	ap-northeast-1	Laden Sie YAML herunter	
Asia Pacific (Osaka)	ap-northeast-3	Laden Sie YAML herunter	
Europa (Frankfurt)	eu-central-1	Laden Sie YAML herunter	
Europa (Irland)	eu-west-1	Laden Sie YAML herunter	

AWS-Region Name	AWS-Region	Quelle ansehen	Stack starten
Europa (London)	eu-west-2	Laden Sie YAML herunter	Launch Stack
Europa (Paris)	eu-west-3	Laden Sie YAML herunter	Launch Stack
Europa (Milan)	eu-south-1	Laden Sie YAML herunter	Launch Stack
Europa (Spain)	eu-south-2	Laden Sie YAML herunter	Launch Stack
Europa (Stockholm)	eu-north-1	Laden Sie YAML herunter	Launch Stack
Südamerika (São Paulo)	sa-east-1	Laden Sie YAML herunter	Launch Stack
AWS GovCloud (US-East)	us-gov-east-1	Laden Sie YAML herunter	Launch Stack
AWS GovCloud (US-West)	us-gov-west-1	Laden Sie YAML herunter	Launch Stack

AWS PCS-Cluster

Ein AWS PCS-Cluster besteht aus den folgenden Komponenten:

- Verwaltete Instanzen der HPC System Scheduler-Software, wie z. B. der Slurm Control Daemon (). `slurmctld`
- Komponenten, die sich in den HPC-Systemplaner integrieren lassen, um EC2 Amazon-Instances bereitzustellen und zu verwalten.
- Komponenten, die in den HPC-Systemplaner integriert sind, um Protokolle und Metriken an Amazon zu übertragen. CloudWatch

Diese Komponenten werden in einem Konto ausgeführt, das von verwaltet wird. AWS Sie arbeiten zusammen, um EC2 Amazon-Instances in Ihrem Kundenkonto zu verwalten. AWS PCS stellt elastische Netzwerkschnittstellen in Ihrem Amazon VPC-Subnetz bereit, um Konnektivität von der Scheduler-Software zu EC2 Amazon-Instances bereitzustellen (z. B. um die Planung von Batch-Jobs auf diesen zu unterstützen und es Benutzern zu ermöglichen, Scheduler-Befehle auszuführen, um diese Jobs aufzulisten und zu verwalten).

Themen

Einen Cluster erstellen in AWS STK.

Dieses Thema bietet einen Überblick über die verfügbaren Optionen und beschreibt, was bei der Erstellung eines Clusters in AWS Parallel Computing Service (AWS PCS) zu beachten ist. Wenn Sie zum ersten Mal einen AWS PCS-Cluster erstellen, empfehlen wir Ihnen, wie folgt vorzugehenErste Schritte mit AWS Parallel Computing Service. Das Tutorial kann Ihnen helfen, ein funktionierendes HPC-System zu erstellen, ohne auf alle verfügbaren Optionen und Systemarchitekturen eingehen zu müssen, die möglich sind.

> **Note**
>
> Nach der Erstellung eines Clusters können Sie viele Konfigurationseinstellungen ändern, ohne Ihre Infrastruktur neu aufbauen zu müssen. Weitere Informationen finden Sie unter Aktualisierung eines Clusters in AWS PCS.

> **Note**
>
> Sie können benutzerdefinierte Slurm-Einstellungen konfigurieren, um erweiterte Planungsrichtlinien und Ressourcenmanagement zu implementieren. Weitere Informationen finden Sie unter Konfiguration benutzerdefinierter Slurm-Einstellungen in AWS STK..

Voraussetzungen

- Eine bestehende VPC und ein Subnetz, die die Anforderungen erfüllenAWS PCS-Netzwerke. Bevor Sie einen Cluster für den Produktionseinsatz bereitstellen, empfehlen wir, dass Sie sich ein umfassendes Verständnis der VPC- und Subnetzanforderungen aneignen. Informationen zum Erstellen einer VPC und eines Subnetzes finden Sie unter. Erstellen Sie eine VPC für Ihr AWS PCS-Cluster
- Ein IAM-Prinzipal mit Berechtigungen zum Erstellen und Verwalten AWS von PCS-Ressourcen. Weitere Informationen finden Sie unter Identity and Access Management für AWS Dienst für parallele Datenverarbeitung.

Erstellen Sie ein AWS PCS-Cluster

Sie können das AWS-Managementkonsole oder verwenden AWS CLI , um einen Cluster zu erstellen.

AWS-Managementkonsole

So erstellen Sie einen Cluster

1. Öffnen Sie die AWS PCS-Konsole unter https://console.aws.amazon.com/pcs/home#/clusters und wählen Sie Create cluster aus.
2. Geben Sie im Abschnitt Cluster-Setup die folgenden Felder ein:

- Clustername — Ein Name für Ihren Cluster. Der Name darf nur alphanumerische Zeichen (wobei die Groß- und Kleinschreibung beachtet werden muss) und Bindestriche enthalten. Er muss mit einem alphabetischen Zeichen beginnen und darf nicht länger als 40 Zeichen sein. Der Name muss innerhalb des AWS-Region und AWS-Konto , in dem Sie den Cluster erstellen, eindeutig sein.
- Scheduler — Wählen Sie einen Scheduler und eine Version aus. Weitere Informationen finden Sie unter Slurm-Versionen in AWS STK..
- Controller-Größe — Wählen Sie eine Größe für Ihren Controller. Dies bestimmt, wie viele gleichzeitige Jobs und Rechenknoten vom AWS PCS-Cluster verwaltet werden können. Sie können die Controller-Größe nur festlegen, wenn der Cluster erstellt wird. Weitere Informationen zur Größenbestimmung finden Sie unterClustergröße in AWS PCS.

3. Wählen Sie im Abschnitt Netzwerk Werte für die folgenden Felder aus:

- Netzwerktyp — Wählen Sie den IP-Adresstyp für Ihren Cluster. Ihr Cluster kann entweder IPv4 oder IPv6 verwenden, aber nicht beide. Die VPC und die Subnetze müssen denselben Netzwerkadresstyp verwenden. Der IP-Adressblock, den Sie für jedes Subnetz verwenden, muss mindestens eine verfügbare Adresse haben. AWS reserviert einige Adressen in jedem Subnetz. Weitere Informationen finden Sie unter Subnetz-CIDR-Blöcke im Amazon-VPC-Benutzerhandbuch.
- VPC — Wählen Sie eine vorhandene VPC, die die AWS PCS-Anforderungen erfüllt. Weitere Informationen finden Sie unter AWS Anforderungen und Überlegungen zu PCS VPC und Subnetzen. Nachdem Sie den Cluster erstellt haben, können Sie seine VPC nicht mehr ändern. Wenn keine VPCs aufgeführt sind, müssen Sie zuerst eine erstellen.
- Subnetz — Alle verfügbaren Subnetze in der ausgewählten VPC werden aufgelistet. Wählen Sie ein Subnetz, das die PCS-Subnetzanforderungen erfüllt. AWS Weitere Informationen finden Sie unter AWS Anforderungen und Überlegungen zu PCS VPC und Subnetzen. Wir empfehlen Ihnen, ein privates Subnetz auszuwählen, um zu verhindern, dass Ihre Scheduler-Endpunkte dem öffentlichen Internet ausgesetzt werden.
- Sicherheitsgruppen — Geben Sie die Sicherheitsgruppe (n) an, die AWS PCS den Netzwerkschnittstellen zuordnen soll, die es für Ihren Cluster erstellt. Sie müssen mindestens eine Sicherheitsgruppe auswählen, die die Kommunikation zwischen Ihrem Cluster und seinen Rechenknoten ermöglicht. Sie können Schnell eine Sicherheitsgruppe erstellen auswählen, damit AWS PCS eine Sicherheitsgruppe mit der erforderlichen Konfiguration in der ausgewählten VPC erstellt, oder Sie können eine vorhandene

Sicherheitsgruppe auswählen. Weitere Informationen finden Sie unter Anforderungen und Überlegungen zur Sicherheitsgruppe.

4. (Optional) Im Abschnitt zur Konfiguration der Slurm-Buchhaltung können Sie die Slurm-Buchhaltung aktivieren und die Abrechnungsparameter festlegen. Weitere Informationen finden Sie unter Slurm-Buchhaltung in AWS STK..
5. (Optional) Im Abschnitt zur Scheduler-Konfiguration können Sie Parameternamen- und Wertepaare hinzufügen, um zusätzliche Slurm-Einstellungen zu konfigurieren. Eine vollständige Liste der unterstützten Parameter finden Sie unter. Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Cluster Sie können auch benutzerdefinierte Cgroup-Einstellungen (siheKonfiguration benutzerdefinierter Cgroup-Einstellungen in AWS STK.) und benutzerdefinierte SlurmDBD-Einstellungen (siehe) konfigurieren. Konfiguration benutzerdefinierter SlurmDBD-Einstellungen in AWS STK.

   Darüber hinaus können Sie auch die folgenden Einstellungen angeben:

   - Scale-down Leerlaufzeit — (Optional) Die Zeit in Sekunden, bis ein Knoten im Leerlauf herunterskaliert wird. Dies gilt standardmäßig für alle Compute-Knotengruppen im Cluster. Für Slurm 25.11 oder höher können einzelne Compute-Knotengruppen diesen Wert mit ihrer eigenen Einstellung für die Scale-down Leerlaufzeit überschreiben.
6. (Optional) Fügen Sie unter Tags beliebige Tags zu Ihrem AWS PCS-Cluster hinzu.
7. Wählen Sie Cluster erstellen. Das Statusfeld wird angezeigt`Creating`, während der AWS PCS den Cluster erstellt. Dieser Vorgang kann einige Minuten dauern.

> **Important**
>
> AWS-Region Pro `Creating` Bundesstaat kann es nur einen Cluster geben AWS-Konto. AWS PCS gibt beim Versuch, einen Cluster zu erstellen, einen Fehler zurück, wenn sich bereits ein Cluster in einem `Creating` Status befindet.

AWS CLI

So erstellen Sie einen Cluster

1. Erstellen Sie den Cluster mit dem folgenden Befehl. Nehmen Sie vor der Ausführung des Befehls die folgenden Ersetzungen vor:

- *region*Ersetzen Sie es durch die ID des Clusters AWS-Region , in dem Sie Ihren Cluster erstellen möchten, z. `us-east-1` B.
- Ersetzen Sie *my-cluster* durch Ihren Cluster-Namen. Der Name darf nur alphanumerische Zeichen (wobei die Groß- und Kleinschreibung beachtet werden muss) und Bindestriche enthalten. Sie muss mit einem alphabetischen Zeichen beginnen und darf nicht länger als 40 Zeichen sein. Der Name muss innerhalb des Clusters AWS-Region und an dem AWS-Konto Ort, an dem Sie den Cluster erstellen, eindeutig sein.
- *25.11*Ersetzen Sie es durch eine unterstützte Version von Slurm.

  > **Note**
  >
  > AWS PCS unterstützt derzeit Slurm 25.11 und 25.05.

- Ersetzen Sie durch eine beliebige *SMALL* unterstützte Clustergröße. Dies bestimmt, wie viele gleichzeitige Jobs und Rechenknoten vom AWS PCS-Cluster verwaltet werden können. Es kann nur festgelegt werden, wenn der Cluster erstellt wird. Weitere Informationen zur Dimensionierung finden Sie unterClustergröße in AWS PCS.
- Ersetzen Sie den Wert für `subnetIds` durch Ihren eigenen. Wir empfehlen Ihnen, ein privates Subnetz auszuwählen, um zu verhindern, dass Ihre Scheduler-Endpunkte dem öffentlichen Internet ausgesetzt werden.
- Geben Sie an`securityGroupIds`, welche Netzwerkschnittstellen AWS PCS den Netzwerkschnittstellen zuordnen soll, die es für Ihren Cluster erstellt. Die Sicherheitsgruppen müssen sich in derselben VPC wie der Cluster befinden. Sie müssen mindestens eine Sicherheitsgruppe auswählen, die die Kommunikation zwischen Ihrem Cluster und seinen Rechenknoten ermöglicht. Weitere Informationen finden Sie unter Anforderungen und Überlegungen zur Sicherheitsgruppe.

```
aws pcs create-cluster --region region \
    --cluster-name my-cluster \
    --scheduler type=SLURM,version=25.11 \
    --size SMALL \
    --networking subnetIds=subnet-ExampleId1,securityGroupIds=sg-ExampleId1
```

- um IPv6 zu verwenden, fügen Sie `networkType=IPV6` es der `--networking` Konfiguration hinzu.

```
  --networking networkType=IPV6,subnetIds=subnet-
ExampleId1,securityGroupIds=sg-ExampleId1
```

- Optional können Sie die `--slurm-configration` Option hinzufügen, das Slurm-Verhalten anzupassen und die Slurm-Konfigurationsoptionen festzulegen. Im folgenden Beispiel wird die Leerlaufzeit beim Herunterfahren auf 60 Minuten (3600 Sekunden) festgelegt, die Slurm-Accounting-Funktion aktiviert und Einstellungen als Wert für angegeben`slurm.conf`. `slurmCustomSettings` Weitere Informationen finden Sie unter Slurm-Buchhaltung in AWS STK..

  > **Note**
  >
  > Accounting wird für Slurm 24.11 oder höher unterstützt.

  > **Note**
  >
  > Die Cluster-Ebene `scaleDownIdleTimeInSeconds` gilt standardmäßig für alle Compute-Knotengruppen im Cluster. Für Slurm 25.11 oder höher können einzelne Compute-Knotengruppen diesen Wert mit ihrer eigenen Einstellung überschreiben. `scaleDownIdleTimeInSeconds`

```
aws pcs create-cluster --region region \
    --cluster-name my-cluster \
    --scheduler type=SLURM,version=25.11 \
    --size SMALL \
    --networking subnetIds=subnet-ExampleId1,securityGroupIds=sg-ExampleId1
    --slurm-configuration
 scaleDownIdleTimeInSeconds=3600,accounting='{mode=STANDARD}',slurmCustomSettings='[{p
```

2. Die Bereitstellung des Clusters kann mehrere Minuten dauern. Sie können den Status Ihres Clusters mit dem folgenden Befehl überprüfen. Fahren Sie erst mit der Erstellung von Warteschlangen oder Compute-Knotengruppen fort, wenn das Statusfeld des Clusters angezeigt wird`ACTIVE`.

```
aws pcs get-cluster --region region --cluster-identifier my-cluster
```

> **Important**
>
> AWS-Region Pro `Creating` AWS-Konto Bundesstaat kann es nur einen Cluster geben. AWS PCS gibt beim Versuch, einen Cluster zu erstellen, einen Fehler zurück, wenn sich bereits ein Cluster in einem `Creating` Status befindet.

Empfohlene nächste Schritte für Ihren Cluster

- Fügen Sie Compute-Knotengruppen hinzu.
- Fügen Sie Warteschlangen hinzu.
- Aktivieren Sie die Protokollierung.

Aktualisierung eines Clusters in AWS PCS

AWS Mit PCS können Sie Clusterkonfigurationen nach der Erstellung über die UpdateCluster API oder Konsole aktualisieren. Sie können die Cluster-Einstellungen ändern, ohne Ihre Infrastruktur neu aufbauen zu müssen, wodurch der Betriebsaufwand reduziert und Unterbrechungen minimiert werden.

Vorteile von Cluster-Updates

Durch die Aktualisierung von AWS PCS-Clustern können Sie die HPC-Infrastruktur ohne Betriebsunterbrechung an neue Anforderungen anpassen. Konfigurationsänderungen dauern Minuten statt der Stunden oder länger, die für die Neuerstellung von Clustern erforderlich sind. Diese Funktion ist wichtig für Produktionsumgebungen, die nur minimale Ausfallzeiten erfordern, und für Teams, die Clustereinstellungen anpassen müssen, wenn sich die Arbeitslastmuster ändern.

Unterstützte Konfigurationsänderungen

Sie können drei Hauptkategorien von Einstellungen ändern:

- Kontoführungskonfiguration — Aktivieren oder deaktivieren Sie die verwaltete Buchhaltung und konfigurieren Sie die Aufbewahrungseinstellungen.

- Verhalten beim Herunterskalieren — Passen Sie den `scaleDownIdleTime` Parameter an, der steuert, wie lange dynamische Instanzen inaktiv bleiben, bevor AWS PCS sie automatisch beendet.
- Benutzerdefinierte Slurm-Einstellungen — Ändern Sie alle unterstützten Slurm-Einstellungen, die auf Cluster-Ebene gelten, einschließlich Prolog, Epilog und. SelectTypeParameters

Einschränkungen

Sie können bestimmte Konfigurationen nach der Clustererstellung nicht ändern. Dazu zählen:

- Konfigurationen von Sicherheitsgruppen
- Auswahl des VPC-Subnetzes
- Cluster-Größe
- Slurm-Version
- Clustername

Diese Einstellungen sind grundlegend für die Architektur des Clusters und erfordern die Erstellung eines neuen Clusters, um sie zu ändern.

Voraussetzungen für Cluster-Updates

Stellen Sie vor dem Aktualisieren eines Clusters sicher, dass die folgenden Bedingungen erfüllt sind:

- Der Cluster muss sich im `SUSPENDED` Status `ACTIVEUPDATE_FAILED`, oder befinden
- Alle zugehörigen Ressourcen (Warteschlangen, Compute-Knotengruppen) müssen sich im Status befinden `ACTIVE`
- Sie müssen über die entsprechenden IAM-Berechtigungen für den Vorgang verfügen UpdateCluster
- Es können keine anderen Aktualisierungsvorgänge ausgeführt werden

Aktualisierungsprozess und Auswirkung auf den Job

Während eines Aktualisierungsvorgangs führen die Rechenknoten weiterhin bestehende Jobs aus, auch wenn der Cluster-Controller kurzzeitig nicht erreichbar ist. In diesem Zeitraum kann das System jedoch keine neuen Auftragseinreichungen annehmen oder Entscheidungen zur Terminplanung treffen.

Sie können Cluster-Updates sowohl über die Konsole als auch über die API-Schnittstelle überwachen. Der Cluster durchläuft während eines Updates die folgenden Zustände:

- `UPDATING`- Aktualisierung läuft
- `ACTIVE`- Das Update wurde erfolgreich abgeschlossen
- `UPDATE_FAILED`- Beim Update ist ein Fehler aufgetreten

Abrechnung bei Updates

Die standardmäßigen Stundengebühren für Ihren AWS PCS-Cluster werden während der Aktualisierungsvorgänge weiterhin berechnet. Wenn Sie einen Cluster aktualisieren, um die Kontoführung zu deaktivieren, wird die Abrechnung für die Abrechnungsfunktion beendet, sobald der Cluster den `UPDATING` Status erreicht. Wenn Sie die Kontoführung aktivieren, beginnt die Abrechnung erst, wenn der Cluster das Update erfolgreich abgeschlossen hat und in den `ACTIVE` Status zurückkehrt.

Themen

- Aktualisieren Sie einen AWS PCS-Cluster
- Häufig gestellte Fragen zur Aktualisierung von Clustern in AWS PCS
- Problembehandlung bei AWS PCS-Cluster-Updates

Aktualisieren Sie einen AWS PCS-Cluster

Gehen Sie wie folgt vor, um die Scheduler-Einstellungen, die Accounting-Konfiguration und die benutzerdefinierten Slurm-Einstellungen auf Ihrem Cluster zu ändern. Weitere Informationen finden Sie unter Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Cluster.

Voraussetzungen

- Der Cluster muss sich im Status `ACTIVEUPDATE_FAILED`, oder befinden `SUSPENDED`
- Alle zugehörigen Ressourcen (Warteschlangen, Compute-Knotengruppen) müssen sich im Status befinden `ACTIVE`
- Es können keine anderen Aktualisierungsvorgänge ausgeführt werden

Verfahren

AWS-Managementkonsole

1. Öffnen Sie die AWS PCS-Konsole unter https://console.aws.amazon.com/pcs/.
2. Klicken Sie im Navigationsbereich auf Cluster.
3. Wählen Sie den zu aktualisierenden Cluster aus.
4. Wählen Sie Bearbeiten aus.
5. Ändern Sie auf der Seite Cluster bearbeiten die gewünschten Einstellungen:
   - Aktualisieren Sie unter Scheduler-Konfiguration die Leerlaufzeit von Scale-down, um zu steuern, wie lange dynamische Instances inaktiv bleiben, bevor sie automatisch beendet werden.
   - Ändern Sie die Parametereinstellungen für Prolog, Epilog und Select nach Bedarf.
   - Aktivieren, deaktivieren oder konfigurieren Sie die Aufbewahrungszeit für Managed Accounting.
   - Fügen Sie unter Zusätzliche Scheduler-Einstellungen benutzerdefinierte Slurm-Einstellungen hinzu, bearbeiten oder entfernen Sie sie. Weitere Informationen zu unterstützten Parametern finden Sie unter. Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Cluster

   > **Note**
   >
   > Felder, die nicht bearbeitet werden können, werden schreibgeschützt angezeigt und zeigen ihre aktuellen Werte an.
6. Wählen Sie „Aktualisieren“, um die Änderungen zu senden.
7. Überwachen Sie den Cluster-Status, der während des Vorgangs als „Aktualisierung“ angezeigt wird. Der Status ändert sich, wenn das Update erfolgreich abgeschlossen wurde.

AWS CLI

1. Öffnen Sie ein Terminal oder eine Befehlszeile.
2. Überprüfen Sie den Clusterstatus mit dem folgenden Befehl:

```
aws pcs get-cluster --cluster-identifier my-cluster
```

3. Senden Sie eine Aktualisierungsanfrage anhand eines der folgenden Beispiele:

   - So aktivieren Sie Managed Accounting:

     ```
     aws pcs update-cluster --cluster-identifier my-cluster \
     --slurm-configuration 'accounting={mode=STANDARD}'
     ```

   - Um eine Slurm Prolog-Einstellung zu aktualisieren:

     ```
     aws pcs update-cluster --cluster-identifier my-cluster \
     --slurm-configuration \
     'SlurmCustomSettings=[{parameterName=Prolog,parameterValue="/path/to/
     prolog.sh"}]'
     ```

   - Um die Leerlaufzeit beim Herunterskalieren zu aktualisieren:

     ```
     aws pcs update-cluster --cluster-identifier my-cluster \
     --slurm-configuration 'scaleDownIdleTimeInSeconds=300'
     ```

4. Überwachen Sie den Aktualisierungsfortschritt, indem Sie den Clusterstatus überprüfen:

   ```
   aws pcs get-cluster --cluster-identifier my-cluster
   ```

Nach einer erfolgreichen Aktualisierungsanforderung gibt der Befehl das Cluster-Objekt mit allen Änderungen zurück. Der Clusterstatus ändert sich von `UPDATING` bis nach `ACTIVE` Abschluss.

Häufig gestellte Fragen zur Aktualisierung von Clustern in AWS PCS

Hier erhalten Sie Antworten auf häufig gestellte Fragen zur Aktualisierung von Clusterkonfigurationen in AWS PCS.

Welche Einstellungen kann ich ändern?

Sie können die Kontoführungskonfiguration (verwaltetes Accounting aktivieren/deaktivieren), das Scale-Down-Verhalten (scaleDownIdleZeitparameter) und alle unterstützten benutzerdefinierten Slurm-Einstellungen, die auf Clusterebene gelten, ändern. Sie können Sicherheitsgruppen, VPC-Subnetze, Clustergröße, Slurm-Version oder Clusternamen nicht ändern.

Kann ich mehrere Updates in die Warteschlange stellen?

Nein. Sie müssen warten, bis der Cluster wieder in den `ACTIVE` Status zurückkehrt, bevor Sie ein weiteres Update einreichen. Alle zugehörigen Ressourcen (Warteschlangen, Compute-Knotengruppen) müssen sich ebenfalls im `ACTIVE` Status befinden.

Kann ich einen Cluster-Aktualisierungsvorgang abbrechen?

Nein, Sie können einen laufenden Cluster-Aktualisierungsvorgang nicht abbrechen.

Kann ich Jobs einreichen, während mein Cluster aktualisiert wird?

Wir empfehlen, dass Sie das Senden von Jobs während der Cluster-Updates vermeiden. Der Slurm-Controller ist während des Aktualisierungsvorgangs möglicherweise nicht verfügbar.

Werden meine Jobs während der Cluster-Updates weiterhin ausgeführt?

Ja, laufende Jobs werden weiterhin auf Rechenknoten ausgeführt, auch wenn der Cluster-Controller während des Aktualisierungsvorgangs kurzzeitig nicht erreichbar ist. Der Jobstatus wird jedoch möglicherweise erst aktualisiert, wenn der Controller wieder verfügbar ist.

Wie wirkt sich die Aktualisierung auf die Abrechnung aus?

Während des Aktualisierungsvorgangs werden weiterhin die Standardgebühren pro Stunde berechnet. Wenn Sie die Kontoführung deaktivieren, wird die Abrechnung beendet, wenn der Cluster in den `UPDATING` Status wechselt. Wenn die Kontoführung aktiviert ist, beginnt die Abrechnung, wenn der Cluster erfolgreich in den `ACTIVE` Status zurückkehrt.

Problembehandlung bei AWS PCS-Cluster-Updates

Dieses Thema hilft Ihnen dabei, häufig auftretende Probleme zu identifizieren und zu lösen, die bei der Aktualisierung von Clusterkonfigurationen auftreten können.

Das Update schlägt mit einem Fehler bei der Kontoführungskonfiguration fehl

Häufige Ursache

Der Cluster wechselt in den `UPDATE_FAILED` Status und die Fehlermeldung weist auf ein Problem mit der Kontoführungskonfiguration hin. Dies tritt normalerweise auf, wenn die Accounting-Konfiguration nicht mit der aktuellen Slurm-Version kompatibel ist oder ungültige Einstellungen enthält.

Auflösung

Überprüfen Sie Ihre Accounting-Einstellungen auf Kompatibilität mit der Slurm-Version Ihres Clusters und reichen Sie eine korrigierte Aktualisierungsanfrage mit gültigen Konfigurationsparametern ein.

Das Update schlägt mit einem Fehler bei den benutzerdefinierten Einstellungen fehl

Häufige Ursache

Der Cluster wechselt in den `UPDATE_FAILED` Status und die Fehlermeldung weist auf ein Problem mit den benutzerdefinierten Slurm-Einstellungen hin. Dies tritt auf, wenn Sie ungültige Slurm-Parameterwerte oder nicht unterstützte Parameterkombinationen angeben.

Auflösung

Überprüfen Sie Ihre benutzerdefinierten Slurm-Einstellungen anhand der unterstützten Parameter und senden Sie eine korrigierte Aktualisierungsanforderung mit gültigen Parameterwerten und Kombinationen.

Aktualisierungsanfrage kann nicht eingereicht werden

Häufige Ursache

Die Aktualisierungsschaltfläche ist in der Konsole deaktiviert oder die API gibt einen Fehler der Stufe 400 zurück. Dies tritt auf, wenn sich der Cluster nicht in einem geeigneten Zustand befindet, die zugehörigen Ressourcen nicht aktiv sind oder wenn in Ihrer Konfiguration Validierungsfehler vorliegen.

Auflösung

Warten Sie, bis der Cluster und alle zugehörigen Ressourcen den `ACTIVE` Status erreicht haben, und überprüfen Sie dann Ihre Konfiguration auf Validierungsfehler, bevor Sie die Aktualisierungsanforderung erneut einreichen.

Validierungsfehler

Häufige Ursache

Der Befehl kehrt sofort mit einem HTTP-Fehler der Stufe 400 und einer beschreibenden Meldung zurück. Dies ist auf ungültige Clusterstatus-, Ressourcenstatus- oder Konfigurationsparameter zurückzuführen.

Auflösung

Beheben Sie den spezifischen Validierungsfehler, der in der Antwort erwähnt wurde, und wiederholen Sie den Aktualisierungsvorgang.

Löschen eines Clusters in AWS PCS

Dieses Thema bietet einen Überblick darüber, wie Sie einen AWS-PCS-Cluster löschen.

Überlegungen beim Löschen eines AWS PCS-Clusters

- Alle mit dem Cluster verknüpften Warteschlangen müssen gelöscht werden, bevor der Cluster gelöscht werden kann. Weitere Informationen finden Sie unter Löschen einer Warteschlange in AWS PCS.
- Alle mit dem Cluster verknüpften Compute-Knotengruppen müssen gelöscht werden, bevor der Cluster gelöscht werden kann. Weitere Informationen finden Sie unter Löschen einer Compute-Knotengruppe in AWS PCS.

Löschen Sie den Cluster

Sie können das AWS-Managementkonsole oder verwenden AWS CLI , um einen Cluster zu löschen.

AWS-Managementkonsole

So löschen Sie einen Cluster

1. Öffnen Sie die AWS PCS-Konsole.
2. Wählen Sie den zu löschenden Cluster aus.
3. Wählen Sie Löschen aus.
4. Das Feld Cluster-Status wird angezeigt`Deleting`. Das kann mehrere Minuten dauern.

AWS CLI

So löschen Sie einen Cluster

1. Verwenden Sie den folgenden Befehl, um einen Cluster mit diesen Ersetzungen zu löschen:
   - Ersetzen Sie *`region-code`* durch den, in dem sich AWS-Region Ihr Cluster befindet.

- *my-cluster*Ersetzen Sie durch den Namen oder die ID Ihres Clusters.

```
aws pcs delete-cluster --region region-code --cluster-identifier my-cluster
```

2. Das Löschen des Clusters kann mehrere Minuten dauern. Sie können den Status Ihres Clusters mit dem folgenden Befehl überprüfen.

```
aws pcs get-cluster --region region-code --cluster-identifier my-cluster
```

Clustergröße in AWS PCS

AWS PCS bietet hochverfügbare und sichere Cluster und automatisiert gleichzeitig wichtige Aufgaben wie Patching, Knotenbereitstellung und Updates.

Wenn Sie einen Cluster erstellen, wählen Sie dessen Größe auf der Grundlage von zwei Faktoren aus:

- Die Anzahl der Rechenknoten, die verwaltet werden
- Die Anzahl der Jobs, die vom Controller zu einem bestimmten Zeitpunkt verfolgt wurden

> **Note**
>
> Die Anzahl der Jobs umfasst laufende, ausstehende und kürzlich abgeschlossene Jobs. Abgeschlossene Jobs werden vom Controller für einen kurzen Zeitraum nachverfolgt, bevor sie gelöscht werden. Bei hohem Auftragsdurchsatz kann dies dazu führen, dass die Gesamtzahl der verfolgten Jobs die Anzahl der aktiven Jobs, die Sie beobachten, übersteigt.

> **Important**
>
> Sie können die Clustergröße nicht ändern, nachdem Sie den Cluster erstellt haben. Wenn Sie die Größe ändern müssen, müssen Sie einen neuen Cluster erstellen.

Größe des Slurm-Clusters	Anzahl der verwalteten Instanzen	Anzahl der vom Controller verfolgten Jobs
Small	Bis zu 32	Bis zu 256
Mittel	Bis zu 512	Bis zu 8192
Large (Groß)	Bis zu 2048	Bis zu 16384

Beispiele

- Wenn Ihr Cluster über bis zu 24 verwaltete Instanzen verfügen und bis zu 100 Jobs ausführen soll, wählen Sie Small.
- Wenn Ihr Cluster über bis zu 24 verwaltete Instanzen verfügen und bis zu 1000 Jobs ausführen soll, wählen Sie Medium.
- Wenn Ihr Cluster bis zu 1000 verwaltete Instanzen haben und bis zu 100 Jobs ausführen soll, wählen Sie Large.
- Wenn Ihr Cluster bis zu 1000 verwaltete Instanzen haben und bis zu 10.000 Jobs ausführen soll, wählen Sie Large.

Arbeiten mit Clustergeheimnissen in AWS PCS

Im Rahmen der Clustererstellung erstellt AWS PCS ein Clustergeheimnis, das für die Verbindung mit dem Job Scheduler auf dem Cluster erforderlich ist. Sie erstellen auch AWS PCS-Compute-Knotengruppen, die Gruppen von Instances definieren, die als Reaktion auf Skalierungsereignisse gestartet werden. AWS PCS konfiguriert Instances, die von diesen Compute-Knotengruppen gestartet werden, mit dem Cluster-Geheimnis, sodass sie eine Verbindung zum Job Scheduler herstellen können. Es gibt Fälle, in denen Sie Slurm-Clients möglicherweise manuell konfigurieren möchten. Beispiele hierfür sind der Aufbau eines persistenten Login-Knotens oder die Einrichtung eines Workflow-Managers mit Job-Management-Funktionen.

AWS PCS speichert das Clustergeheimnis als verwaltetes Geheimnis mit dem Präfix `pcs!` in AWS Secrets Manager. Die Kosten für das Secret sind in der Gebühr für die Nutzung von AWS PCS enthalten. Sie können Clustergeheimnisse rotieren, AWS Secrets Manager um die Einhaltung der Sicherheitsbestimmungen zu gewährleisten und potenzielle Sicherheitslücken zu beheben.

Themen

Wird verwendet AWS Secrets Manager , um das Cluster-Geheimnis zu finden

AWS-Managementkonsole

1. Navigieren Sie zur Secrets Manager Manager-Konsole.
2. Wählen Sie Secrets und suchen Sie dann nach dem `pcs!` Präfix.

> **Note**
>
> Ein AWS PCS-Clustergeheimnis hat einen Namen in der Form`pcs!slurm-secret-`*`cluster-id`*, in der die AWS PCS-Cluster-ID *`cluster-id`* steht.

AWS CLI

Jedes geheime AWS PCS-Clustergeheimnis ist ebenfalls mit gekennzeichnet`aws:pcs:`*`cluster-id`*. Sie können die geheime ID für einen Cluster mit dem folgenden Befehl abrufen. Nehmen Sie diese Ersetzungen vor, bevor Sie den Befehl ausführen:

- *`region`*Ersetzen Sie es durch das AWS-Region , in dem Sie Ihren Cluster erstellen möchten, z. B. `us-east-1`
- *`cluster-id`*Ersetzen Sie es durch die ID des AWS PCS-Clusters, für den Sie den Clusterschlüssel finden möchten.

```
aws secretsmanager list-secrets \
    --region region  \
    --filters Key=tag-key,Values=aws:pcs:cluster-id \
            Key=tag-value,Values=cluster-id
```

Verwenden Sie AWS PCS, um das Cluster-Geheimnis zu finden

Sie können das verwenden AWS CLI , um den ARN für ein AWS PCS-Clustergeheimnis zu finden. Geben Sie den folgenden Befehl ein und nehmen Sie die folgenden Ersetzungen vor:

- *region*Ersetzen Sie durch den AWS-Region , in dem Sie Ihren Cluster erstellen möchten, z. B. `us-east-1`
- *my-cluster*Ersetzen Sie durch den Namen oder die Kennung für Ihren Cluster.

```
aws pcs get-cluster --region region --cluster-identifier my-cluster
```

Die folgende Beispielausgabe stammt aus dem `get-cluster` Befehl. Sie können `secretArn` und `secretVersion` zusammen verwenden, um das Geheimnis zu ermitteln.

```
{
    "cluster": {
        "name": "get-started",
        "id": "pcs_123456abcd",
        "arn": "arn:aws:pcs:us-east-1:111122223333:cluster/pcs_123456abcd",
        "status": "ACTIVE",
        "createdAt": "2024-12-17T21:03:52+00:00",
        "modifiedAt": "2024-12-17T21:03:52+00:00",
        "scheduler": {
            "type": "SLURM",
            "version": "25.11"
        },
        "size": "SMALL",
        "slurmConfiguration": {
            "authKey": {
                "secretArn": "arn:aws:secretsmanager:us-east-1:111122223333:secret:pcs!
slurm-secret-pcs_123456abcd-a12ABC",
                "secretVersion": "ef232370-d3e7-434c-9a87-ec35c1987f75"
            }
        },
        "networking": {
            "subnetIds": [
                "subnet-0123456789abcdef0"
            ],
            "securityGroupIds": [
                "sg-0123456789abcdef0"
            ]
```

```
        },
        "endpoints": [
            {
                "type": "SLURMCTLD",
                "privateIpAddress": "10.3.149.220",
                "port": "6817"
            }
        ]
    }
}
```

Holen Sie sich das Geheimnis des Slurm-Clusters

Sie können Secrets Manager verwenden, um die aktuelle Base64-kodierte Version eines Slurm-Cluster-Secrets abzurufen. Das folgende Beispiel verwendet die. AWS CLI Nehmen Sie die folgenden Ersetzungen vor, bevor Sie den Befehl ausführen.

- *region*Ersetzen Sie es durch das AWS-Region , in dem Sie Ihren Cluster erstellen möchten, z. B. `us-east-1`
- *secret-arn*Ersetzen Sie durch das `secretArn` aus einem AWS PCS-Cluster.

```
aws secretsmanager get-secret-value \
    --region region \
    --secret-id 'secret-arn' \
    --version-stage AWSCURRENT \
    --query 'SecretString' \
    --output text
```

Hinweise zur Verwendung des Slurm-Clustergeheimnisses finden Sie unterVerwenden eigenständiger Instanzen als AWS PCS-Anmeldeknoten.

Berechtigungen

Sie verwenden einen IAM-Principal, um das geheime Slurm-Clustergeheimnis abzurufen. Der IAM-Principal muss berechtigt sein, das Geheimnis zu lesen. Weitere Informationen finden Sie im AWS Identity and Access Management Benutzerhandbuch unter Begriffe und Konzepte für Rollen.

Die folgende Beispiel-IAM-Richtlinie ermöglicht den Zugriff auf ein Beispiel für ein Clustergeheimnis.

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowSecretValueRetrievalAndVersionListing",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue",
                "secretsmanager:ListSecretVersionIds"
            ],
            "Resource": "arn:aws:secretsmanager:us-east-1:012345678901:secret:pcs!
slurm-secret-s3431v9rx2-FN7tJF"
        }
    ]
}
```

Rotierende Clustergeheimnisse in AWS PCS

Verwenden Sie AWS Secrets Manager Managed Rotation, um Cluster-Geheimnisse in AWS PCS rotieren zu lassen. Die regelmäßige Rotation von Geheimnissen ist eine bewährte Sicherheitsmethode zur Aufrechterhaltung eines hohen Sicherheitsniveaus in HPC-Umgebungen. Diese Funktion ermöglicht es Ihnen, branchenübliche Compliance-Standards wie HIPAA und FedRAMP zu erfüllen, die eine regelmäßige Rotation von Anmeldeinformationen vorschreiben.

Das Clustergeheimnis dient zwei Zwecken: zur Authentifizierung von Rechenknoten, die dem Cluster beitreten, und als JWT-Schlüssel für die Slurm-REST-API-Authentifizierung. Bei Rotation wirken sich beide Aspekte gleichzeitig aus.

So funktioniert die geheime Cluster-Rotation

Bereiten Sie sich manuell vor, um die Cluster-Stabilität während der geheimen Rotation aufrechtzuerhalten:

1. Vorbereitung — Skalieren Sie alle Rechenknotengruppen auf eine Kapazität von 0 und stellen Sie sicher, dass keine Jobs ausgeführt werden
2. Rotation — Initiieren Sie die Rotation über die Secrets Manager Manager-Konsole oder API
3. Überwachung — Verfolgen Sie den Fortschritt anhand von CloudTrail Ereignissen
4. Wiederherstellung — Skalieren Sie die Rechenknotengruppen wieder auf die gewünschte Kapazität

Während der Rotation bleibt Ihr Cluster unverändert `ACTIVE` und die Abrechnung läuft normal weiter. Der Vorgang dauert in der Regel einige Minuten.

Anforderungen und Einschränkungen

Bevor Sie Clustergeheimnisse rotieren, müssen Sie die folgenden Anforderungen erfüllen:

- Der Cluster muss den `UPDATE_FAILED` Status „`ACTIVE`oder“ haben
- Die IAM-Rolle muss über eine entsprechende Berechtigung verfügen `secretsmanager:RotateSecret`
- Alle Compute-Knotengruppen müssen auf eine Kapazität von 0 skaliert werden
- Stoppen Sie alle Jobs vor der Rotation

Einschränkungen:

- Für jede Rotation ist eine manuelle Vorbereitung erforderlich
- Bestehende JWT-Token werden ungültig und müssen erneut ausgestellt werden
- BYO-Anmeldeknoten müssen nach der Rotation manuell geheim aktualisiert werden

Topics

Rotieren Sie ein Clustergeheimnis in AWS PCS

Wechseln Sie Ihr Clustergeheimnis, um die Sicherheitsanforderungen zu erfüllen und potenzielle Sicherheitslücken zu vermeiden. Dieser Vorgang erfordert, dass Ihr Cluster in den Wartungsmodus versetzt wird.

Voraussetzungen

- IAM-Rolle mit Genehmigung `secretsmanager:RotateSecret`
- Cluster in `ACTIVE` oder im Bundesstaat `UPDATE_FAILED`

Verfahren

1. Informieren Sie die Cluster-Benutzer über das bevorstehende Wartungsfenster.
2. Versetzen Sie den Cluster in den Wartungsmodus, indem Sie alle Rechenknotengruppen auf eine Kapazität von 0 skalieren.
    a. Verwenden Sie die UpdateComputeNodeGroup API, maxInstanceCount um minInstanceCount sowohl als auch für alle Compute-Knotengruppen auf 0 zu setzen.
    b. Warten Sie, bis alle Knoten gestoppt sind.
    c. Optional: Entleeren Sie die Scheduler-Warteschlangen mit Slurm-Befehlen, bevor Sie die Kapazität für eine reibungslose Auftragsabwicklung beenden.
3. Initiieren Sie die Rotation über Secrets Manager.
    - Konsolenmethode:
        - Navigieren Sie zu Secrets Manager, wählen Sie Ihr Clustergeheimnis aus und wählen Sie Rotate Secret aus.
    - API-Methode:
        - Verwenden Sie die Secrets Manager `rotate-secret` Manager-API.
4. Überwachen Sie den Fortschritt der Rotation.
    a. Verfolgen Sie den Fortschritt anhand von CloudTrail Ereignissen.
    b. Überprüfen Sie `lastRotatedDate` dies entweder über die Secrets Manager Manager-Konsole oder die `secretsmanager:describeSecret` API.
    c. Warten Sie auf `RotationSucceeded` unser `RotationFailed` CloudTrail Ereignis.
5. Stellen Sie nach erfolgreicher Rotation die Clusterkapazität wieder her.
    a. Verwenden Sie die UpdateComputeNodeGroup API, um Knotengruppen auf die gewünschte min/max Kapazität zurückzusetzen.
    b. Für AWS PCS-verwaltete Anmeldeknoten: Keine zusätzlichen Maßnahmen erforderlich.
    c. Für BYO-Anmeldeknoten:
        i. Connect zu Anmeldeknoten her.
        ii. Aktualisiere `/etc/slurm/slurm.key` mit dem neuen Secret von Secrets Manager.
        iii. Starte den Slurm Auth and Cred Kiosk Daemon (sackd) neu.

Häufig gestellte Fragen zur geheimen Cluster-Rotation in AWS PCS

Hier finden Sie Antworten auf häufig gestellte Fragen zur geheimen Cluster-Rotation in AWS PCS.

Was ist ein geheimer Clusterschlüssel?

Ein Clustergeheimnis ist ein sicherer Berechtigungsnachweis, der eine sichere Kommunikation zwischen dem Slurm-Controller und den AWS PCS-Rechenknoten ermöglicht. Es dient auch als JSON Web Token (JWT) -Schlüssel für die Slurm-REST-API-Authentifizierung.

Was ist der Unterschied zwischen Cluster-Secret und JWT-Schlüssel?

In AWS PCS sind das Clustergeheimnis und der JWT-Schlüssel dieselbe Ressource, die unterschiedlichen Zwecken dient. Das Clustergeheimnis authentifiziert die interne Kommunikation von Slurm, während der JWT-Schlüssel Token für die REST-API-Authentifizierung signiert. Bei Rotation sind beide Aspekte gleichzeitig betroffen.

Wie lange dauert die Rotation?

Der Rotationsvorgang dauert in der Regel einige Minuten. Ihr Cluster bleibt im Status AKTIV und die Abrechnung läuft während der Rotation normal weiter.

Kann ich automatische Rotationen planen?

Sie können die geplante Rotation in Secrets Manager aktivieren. Die erste Version erfordert jedoch vor jeder Rotation eine manuelle Vorbereitung (Skalierung der Knotengruppen auf 0).

Funktionieren meine vorhandenen JWT-Token nach der Rotation noch?

Nein, bestehende JWT-Token werden nach der Rotation ungültig. Geben Sie neue Token für REST-API-Clients aus.

Wo finde ich mein Clustergeheimnis?

Sie finden Ihr Clustergeheimnis in der Secrets Manager-Konsole oder über die AWS PCS-Konsole. Eine ausführliche Anleitung finden Sie unter Wird verwendet AWS Secrets Manager , um das Cluster-Geheimnis zu finden undVerwenden Sie AWS PCS, um das Cluster-Geheimnis zu finden.

Warum erfordert die Rotation die Skalierung von Knotengruppen auf 0?

Für die Rotation sind keine laufenden Instances erforderlich, um die Cluster-Stabilität während des geheimen Aktualisierungsprozesses zu gewährleisten. Dadurch werden Authentifizierungskonflikte zwischen alten und neuen Geheimnissen vermieden.

Welche Compliance-Anforderungen unterstützt diese Funktion?

Diese Funktion ermöglicht es AWS PCS, branchenübliche Compliance-Standards wie HIPAA und FedRAMP zu erfüllen, die im Rahmen ihrer Sicherheitskontrollen eine regelmäßige Rotation von Anmeldeinformationen vorschreiben.

Fehlerbehebung bei der geheimen Cluster-Rotation in AWS PCS

Die Rotation des geheimen Clusters schlägt fehl, wenn die Umgebung nicht ordnungsgemäß vorbereitet ist. Die häufigste Ursache sind aktive Instanzen in Ihrem Cluster. Um Ausfälle zu verhindern:

1. Stellen Sie für alle Knotengruppen die Kapazität 0 ein.
2. Warten Sie, bis die Knoten gestoppt sind.
3. Stellen Sie sicher, dass sich Ihr Cluster nicht in den folgenden Zuständen befindet: `CREATE_FAILED` `DELETE_FAILED` `RESUMING`,`SUSPENDING`,, oder`SUSPENDED`.

Wenn die Rotation fehlschlägt:

- Es erscheint ein RotationFailed CloudTrail Ereignis
- Das Clustergeheimnis bleibt unverändert
- Einzelheiten finden Sie in CloudTrail der RotationFailed Veranstaltung
- Schließe alle Vorbereitungsschritte für eine erfolgreiche Rotation ab

AWS PCS-Compute-Knotengruppen

Eine AWS PCS-Rechenknotengruppe ist eine logische Sammlung von Knoten (Amazon EC2 EC2-Instances). Diese Knoten können für die Ausführung von Rechenjobs sowie für den interaktiven, Shell-basierten Zugriff auf ein HPC-System verwendet werden. Eine Compute-Knotengruppe besteht aus Regeln für die Erstellung von Knoten, einschließlich der zu verwendenden Amazon EC2 EC2-Instance-Typen, der Anzahl der auszuführenden Instances, ob Spot-Instances oder On-demand Instances verwendet werden sollen, welche Subnetze und Sicherheitsgruppen verwendet werden sollen und wie jede Instance beim Start konfiguriert wird. Wenn diese Regeln aktualisiert werden, aktualisiert AWS PCS die der Rechenknotengruppe zugewiesenen Ressourcen entsprechend.

⚠ SMT ist auf Rechenknoten deaktiviert

AWS PCS deaktiviert Simultanes Multithreading (SMT), auch bei Intel-Prozessoren genannt, Hyper-Threading auf allen Rechenknoteninstanzen beim Bootstrap. Dies ist nicht konfigurierbar. Bei SMT-capable Instance-Typen ist jede vCPU einem dedizierten physischen Kern und nicht einem Hardware-Thread zugeordnet. Das bedeutet, dass die Gesamtzahl der vCPUs die Hälfte der Standardanzahl für den Instanztyp beträgt, aber jede vCPU exklusiven Zugriff auf den gesamten Kern hat. Ein Instance-Typ, der 96 vCPUs bewirbt, hat beispielsweise 48 nutzbare Kerne auf AWS PCS-Rechenknoten. Instance-Typen, die SMT nicht unterstützen, wie Graviton (Arm), sind davon nicht betroffen.
Bei den meisten rechenintensiven HPC-Workloads ist die Leistung bei deaktiviertem SMT gleichwertig oder besser. Durch die Deaktivierung von Hyperthreading werden Ressourcenkonflikte zwischen gleichgeordneten Threads vermieden und jeder physische Kern hat exklusiven Zugriff auf seinen Cache und seine Ausführungseinheiten. Dies ist in HPC-Umgebungen üblich.

Themen

Erstellen einer Compute-Knotengruppe in AWS STK.

Dieses Thema bietet einen Überblick über die verfügbaren Optionen und beschreibt, was zu beachten ist, wenn Sie eine Rechenknotengruppe in AWS Parallel Computing Service (AWS PCS) erstellen. Wenn Sie zum ersten Mal eine Rechenknotengruppe in AWS PCS erstellen, empfehlen wir Ihnen, das Tutorial unter zu befolgenErste Schritte mit AWS Parallel Computing Service. Das Tutorial kann Ihnen helfen, ein funktionierendes HPC-System zu erstellen, ohne auf alle verfügbaren Optionen und Systemarchitekturen eingehen zu müssen, die möglich sind.

> **Note**
>
> Sie können benutzerdefinierte Slurm-Einstellungen für Compute-Knotengruppen konfigurieren, um die Ressourcennutzung und das Verhalten auf Knotenebene zu kontrollieren. Weitere Informationen finden Sie unter Konfiguration benutzerdefinierter Slurm-Einstellungen in AWS STK..

> **Important**
>
> AWS PCS benötigt derzeit einen Kernel mit IPv4-Unterstützung für die Kommunikation mit lokalen Knoten, auch wenn Sie AWS PCS in einem Netzwerk verwenden. IPv6-only Weitere Informationen finden Sie unter Benutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS.

Voraussetzungen

- Ausreichende Servicekontingenten, um die gewünschte Anzahl von EC2-Instances in Ihrem zu starten. AWS-Region Sie können das verwenden AWS-Managementkonsole, um eine Erhöhung Ihrer Servicekontingenten zu überprüfen und zu beantragen.
- Eine bestehende VPC und Subnetze, die die AWS PCS-Netzwerkanforderungen erfüllen. Wir empfehlen, dass Sie sich gründlich mit diesen Anforderungen vertraut machen, bevor Sie einen Cluster für die Produktion bereitstellen. Weitere Informationen finden Sie unter AWS Anforderungen und Überlegungen zu PCS VPC und Subnetzen. Sie können auch eine CloudFormation Vorlage verwenden, um eine VPC und Subnetze zu erstellen. AWS stellt ein HPC-Rezept für die Vorlage bereit. CloudFormation Weitere Informationen finden Sie unter aws-hpc-recipes on. GitHub

- Ein IAM-Instanzprofil mit Berechtigungen zum Aufrufen der AWS `RegisterComputeNodeGroupInstance` PCS-API-Aktion und zum Zugriff auf alle anderen AWS Ressourcen, die für Ihre Knotengruppen-Instances erforderlich sind. Weitere Informationen finden Sie unter IAM-Instanzprofile für AWS Dienst für parallele Datenverarbeitung.
- Eine Startvorlage für Ihre Knotengruppen-Instances. Weitere Informationen finden Sie unter Verwenden von Amazon EC2 EC2-Startvorlagen mit AWS PCS.
- Um eine Compute-Knotengruppe zu erstellen, die Amazon EC2-Spot-Instances verwendet, müssen Sie die AWSServiceRoleForEC2Spotserviceverknüpfte Rolle in Ihrer haben. AWS-Konto Weitere Informationen finden Sie unter Amazon EC2 Spot-Rolle für AWS STK..

Erstellen Sie eine Rechenknotengruppe in AWS STK.

Sie können eine Compute-Knotengruppe mit dem AWS-Managementkonsole oder dem erstellen AWS CLI.

AWS-Managementkonsole

Um Ihre Compute-Knotengruppe mithilfe der Konsole zu erstellen

1. Öffnen Sie die AWS PCS-Konsole.
2. Wählen Sie den Cluster aus, in dem Sie eine Compute-Knotengruppe erstellen möchten. Navigieren Sie zu Compute-Knotengruppen und wählen Sie Create aus.
3. Geben Sie im Abschnitt Konfiguration der Compute-Knotengruppe einen Namen für Ihre Knotengruppe ein. Der Name darf nur alphanumerische Zeichen und Bindestriche enthalten, bei denen Groß- und Kleinschreibung beachtet wird. Er muss mit einem alphabetischen Zeichen beginnen und darf nicht länger als 25 Zeichen sein. Der Name muss innerhalb des Clusters eindeutig sein.
4. Geben Sie unter Computerkonfiguration die folgenden Werte ein, oder wählen Sie sie aus:

    a. EC2-Startvorlage — Wählen Sie eine benutzerdefinierte Startvorlage aus, die für diese Knotengruppe verwendet werden soll. Startvorlagen können verwendet werden, um Netzwerkeinstellungen wie Subnetz und Sicherheitsgruppen, Überwachungskonfiguration und Speicher auf Instanzebene anzupassen. Falls Sie noch keine Startvorlage vorbereitet haben, erfahren Sie unter, wie Verwenden von Amazon EC2 EC2-Startvorlagen mit AWS PCS Sie eine erstellen.

> **Important**
>
> AWS PCS erstellt eine verwaltete Startvorlage für jede Rechenknotengruppe. Diese sind benannt`pcs-`*`identifier`*`-do-not-delete`. Wählen Sie diese nicht aus, wenn Sie eine Compute-Knotengruppe erstellen oder aktualisieren, da die Knotengruppe sonst nicht richtig funktioniert.

b. Version der EC2-Startvorlage — Sie müssen eine Version Ihrer benutzerdefinierten Startvorlage auswählen. Wenn Sie die Version später ändern, müssen Sie die Compute-Knotengruppe aktualisieren, um Änderungen in der Startvorlage zu erkennen. Weitere Informationen finden Sie unter Aktualisierung eines AWS PCS-Compute-Knotengruppe.

c. AMI-ID — Wenn Ihre Startvorlage keine AMI-ID enthält oder wenn Sie den Wert in der Startvorlage überschreiben möchten, geben Sie hier eine AMI-ID ein. Beachten Sie, dass das für die Knotengruppe verwendete AMI mit AWS PCS kompatibel sein muss. Sie können auch ein Beispiel-AMI auswählen, das von bereitgestellt wird AWS. Weitere Informationen zu diesem Thema finden Sie unterAmazon Machine Images (AMIs) für AWS STK..

d. IAM-Instanzprofil — Wählen Sie ein Instanzprofil für die Knotengruppe aus. Ein Instanzprofil gewährt der Instanz Berechtigungen für den sicheren Zugriff auf AWS Ressourcen und Dienste. Wenn Sie noch kein Basisprofil vorbereitet haben, können Sie „Basisprofil erstellen“ auswählen, damit AWS PCS eines für Sie mit der Mindestrichtlinie erstellt, oder sehen Sie nachIAM-Instanzprofile für AWS Dienst für parallele Datenverarbeitung.

e. Subnetze — Wählen Sie ein oder mehrere Subnetze in der VPC aus, in der Ihr AWS PCS-Cluster bereitgestellt wird. Wenn Sie mehrere Subnetze auswählen, ist die EFA-Kommunikation zwischen den Knoten nicht verfügbar, und die Kommunikation zwischen Knoten in verschiedenen Subnetzen kann zu einer erhöhten Latenz führen. Stellen Sie sicher, dass die Subnetze, die Sie hier angeben, mit denen übereinstimmen, die Sie in der EC2-Startvorlage definiert haben.

f. Instances — Wählen Sie einen oder mehrere Instance-Typen aus, um Skalierungsanforderungen in der Knotengruppe zu erfüllen. Alle Instance-Typen müssen dieselbe Prozessorarchitektur (x86_64 oder arm64) und dieselbe Anzahl von vCPUs haben. Wenn die Instanzen GPUs haben, müssen alle Instanztypen dieselbe Anzahl von GPUs haben.

g. Skalierungskonfiguration — Geben Sie die Mindest- und Höchstanzahl von Instanzen für die Knotengruppe an. Stellen Sie das Minimum (min) gleich dem Maximum (max.) für die statische Kapazität ein (z. B. 5 min, 5 max.). Stellen Sie das Minimum für eine volldynamische Skalierung auf 0 ein (z. B. 0 min, 10 max). Bei Slurm 24.05 oder höher können Sie das Minimum für gemischte Kapazitäten auf einen Wert über 0 und unter den Höchstwert setzen. Dadurch wird eine Basisanzahl von Instanzen beibehalten und bei Bedarf hochskaliert (z. B. 2 Minuten, maximal 10).

5. (Optional) Geben Sie unter Zusätzliche Einstellungen Folgendes an:

   a. Kaufoption — Wählen Sie On-Demand Instances, Spot-Instances, eine unterbrechbare Kapazitätsreservierung oder einen vorhandenen Kapazitätsblock aus. Wählen Sie aus, On-Demandob Sie eine On-Demand Kapazitätsreservierung (ODCR) verwenden möchten. Weitere Informationen finden Sie unter Verwendung ODCRs mit AWS PCS. Wählen Sie „Unterbrechbare Kapazitätsreservierung“, um ein gemeinsam genutztes unterbrechbares ODCR () zu verwenden. I-ODCR Weitere Informationen finden Sie unter Verwenden I-ODCRs mit AWS STK.. Wählen Sie Capacity Block, um einen vorhandenen Amazon EC2 EC2-Kapazitätsblock für die ML-Reservierung zu verwenden. Weitere Informationen finden Sie unter Verwenden von Amazon EC2 EC2-Kapazitätsblöcken für ML mit AWS PCS.

   b. Zuweisungsstrategie — Wenn Sie die Spot-Kaufoption ausgewählt haben, können Sie angeben, wie Spot-Kapazitätspools beim Starten von Instances in der Knotengruppe ausgewählt werden. Weitere Informationen finden Sie unter Zuweisungsstrategien für Spot-Instances im Amazon Elastic Compute Cloud-Benutzerhandbuch. Diese Option hat keine Auswirkung, wenn Sie die On-demand Kaufoption ausgewählt haben.

6. (Optional) Im Abschnitt „Scheduler-Konfiguration“ können Sie Folgendes angeben:

   - Scale-down Leerlaufzeit — (Optional) Die Zeit in Sekunden, bis ein inaktiver Knoten in dieser Knotengruppe herunterskaliert wird. Wenn nicht festgelegt, wird der Wert auf Clusterebene verwendet. Für diese Einstellung ist Slurm Version 25.11 oder höher erforderlich.

7. (Optional) Im Bereich für Slurmbenutzerdefinierte Einstellungen können Sie Parameternamen- und Wertepaare hinzufügen, um zusätzliche Slurm-Einstellungen zu konfigurieren. Eine vollständige Liste der unterstützten Parameter finden Sie unterBenutzerdefinierte Slurm-Einstellungen für AWS PCS-Compute-Knotengruppen.

8. (Optional) Fügen Sie unter Tags beliebige Tags zu Ihrer Compute-Knotengruppe hinzu.

9. Wählen Sie Compute-Knotengruppe erstellen aus. Im Feld Status wird angezeigt, `Creating` während AWS PCS die Knotengruppe bereitstellt. Dies kann mehrere Minuten dauern.

Als nächster Schritt wird empfohlen

- Fügen Sie Ihre Knotengruppe zu einer Warteschlange in AWS PCS hinzu, damit sie Jobs verarbeiten kann.

AWS CLI

So erstellen Sie Ihre Compute-Knotengruppe mit AWS CLI

Erstellen Sie Ihre Warteschlange mit dem folgenden Befehl. Nehmen Sie vor der Ausführung des Befehls die folgenden Ersetzungen vor:

1. *`region`*Ersetzen Sie es durch die ID des AWS-Region , in dem Sie Ihren Cluster erstellen möchten, z. `us-east-1` B.
2. *`my-cluster`*Ersetzen Sie durch den Namen oder `clusterId` Ihres Clusters.
3. *`my-node-group`*Ersetzen Sie es durch den Namen für Ihre Compute-Knotengruppe. Der Name darf nur alphanumerische Zeichen (wobei die Groß- und Kleinschreibung beachtet werden muss) und Bindestriche enthalten. Er muss mit einem alphabetischen Zeichen beginnen und darf nicht länger als 25 Zeichen sein. Der Name muss innerhalb des Clusters eindeutig sein.
4. *`subnet-ExampleID1`*Ersetzen Sie durch eine oder mehrere Subnetz-IDs aus Ihrer Cluster-VPC.
5. *`lt-ExampleID1`*Ersetzen Sie es durch die ID für Ihre benutzerdefinierte Startvorlage. Falls Sie noch keine vorbereitet haben, erfahren Verwenden von Amazon EC2 EC2-Startvorlagen mit AWS PCS Sie unter, wie Sie eine erstellen.

> **Important**
>
> AWS PCS erstellt für jede Rechenknotengruppe eine verwaltete Startvorlage. Diese sind benanntpcs-*`identifier`*-do-not-delete. Wählen Sie diese nicht aus, wenn Sie eine Compute-Knotengruppe erstellen oder aktualisieren, da die Knotengruppe sonst nicht richtig funktioniert.

6. *launch-template-version*Ersetzen Sie sie durch eine bestimmte Version der Startvorlage. AWS PCS ordnet Ihre Knotengruppe dieser spezifischen Version der Startvorlage zu.
7. *arn:InstanceProfile*Ersetzen Sie es durch den ARN Ihres IAM-Instanzprofils. Falls Sie noch keinen vorbereitet haben, finden Sie weitere Verwenden von Amazon EC2 EC2-Startvorlagen mit AWS PCS Informationen unter.
8. Ersetzen Sie *min-instances* und durch *max-instances* ganzzahlige Werte. Stellen Sie das Minimum (min) gleich dem Maximum (max) für die statische Kapazität ein (z. B. 5 min, 5 max.). Stellen Sie das Minimum für eine volldynamische Skalierung auf 0 ein (z. B. 0 min, 10 max). Bei Slurm 24.05 oder höher können Sie das Minimum für gemischte Kapazitäten auf einen Wert über 0 und unter den Höchstwert setzen. Dadurch wird eine Basisanzahl von Instanzen beibehalten und bei Bedarf hochskaliert (z. B. 2 Minuten, maximal 10).
9. Durch einen *t3.large* anderen Instanztyp ersetzen. Sie können weitere Instanztypen hinzufügen, indem Sie eine Liste mit `instanceType` Einstellungen angeben. Beispiel, *--instance-configs instanceType=c6i.16xlarge instanceType=c6a.16xlarge*. Alle Instance-Typen müssen dieselbe Prozessorarchitektur (x86_64 oder arm64) und dieselbe Anzahl von vCPUs haben. Wenn die Instanzen GPUs haben, müssen alle Instanztypen dieselbe Anzahl von GPUs haben.

```
aws pcs create-compute-node-group --region region \
    --cluster-identifier my-cluster \
    --compute-node-group-name my-node-group \
    --subnet-ids subnet-ExampleID1 \
    --custom-launch-template id=lt-ExampleID1,version='launch-template-version' \
    --iam-instance-profile-arn=arn:InstanceProfile \
    --scaling-config minInstanceCount=min-instances,maxInstanceCount=max-instance \
    --instance-configs instanceType=t3.large
```

Example— Erstellen einer Compute-Knotengruppe mit benutzerdefinierten Slurm-Einstellungen

```
aws pcs create-compute-node-group --region region \
    --cluster-identifier my-cluster \
    --compute-node-group-name my-node-group \
    --subnet-ids subnet-ExampleID1 \
    --custom-launch-template id=lt-ExampleID1,version='launch-template-version' \
    --iam-instance-profile-arn=arn:InstanceProfile \
    --scaling-config minInstanceCount=min-instances,maxInstanceCount=max-instance \
    --instance-configs instanceType=t3.large \
```

```
    --slurm-configuration \
    'slurmCustomSettings=[{parameterName=Features,parameterValue="gpu,nvme"}]'
```

Weitere Informationen finden Sie unter Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Compute-Knotengruppen.

Example— Erstellung einer Rechenknotengruppe mit einer nach unten skalierten Leerlaufzeit

```
aws pcs create-compute-node-group --region region \
    --cluster-identifier my-cluster \
    --compute-node-group-name my-gpu-nodes \
    --subnet-ids subnet-ExampleID1 \
    --custom-launch-template id=lt-ExampleID1,version='1' \
    --iam-instance-profile-arn=arn:InstanceProfile \
    --scaling-config minInstanceCount=0,maxInstanceCount=10 \
    --instance-configs instanceType=p4d.24xlarge \
    --slurm-configuration scaleDownIdleTimeInSeconds=300
```

Die `scaleDownIdleTimeInSeconds` Einstellung auf der Ebene der Compute-Knotengruppe hat Vorrang vor dem Wert auf Clusterebene für Knoten in dieser Gruppe. Für diese Einstellung ist Slurm Version 25.11 oder höher erforderlich.

Es gibt mehrere optionale Konfigurationseinstellungen, die Sie dem `create-compute-node-group` Befehl hinzufügen können.

- Sie können angeben, `--amiId` ob Ihre benutzerdefinierte Startvorlage keinen Verweis auf ein AMI enthält oder ob Sie diesen Wert überschreiben möchten. Beachten Sie, dass das für die Knotengruppe verwendete AMI mit AWS PCS kompatibel sein muss. Sie können auch ein Beispiel-AMI auswählen, das von bereitgestellt wird AWS. Weitere Informationen zu diesem Thema finden Sie unterAmazon Machine Images (AMIs) für AWS STK..
- Dient `--purchase-option` zur Auswahl der Art und Weise, wie AWS PCS EC2-Instances für Ihre Compute-Knotengruppe kauft. On-Demand ist die Standardeinstellung.
  - `ONDEMAND`— Verwenden Sie On-Demand Instanzen. Wählen Sie diese Option auch, wenn Sie eine On-Demand Kapazitätsreservierung (ODCR) verwenden möchten. Weitere Informationen finden Sie unter Verwendung ODCRs mit AWS PCS.
  - `SPOT`— Verwenden Sie Spot-Instances. Wenn Sie Spot-Instances wählen, können Sie `--allocation-strategy` damit auch definieren, wie AWS PCS Spot-Kapazitätspools auswählt, wenn Instances in der Knotengruppe gestartet werden. Weitere Informationen

finden Sie unter Zuweisungsstrategien für Spot-Instances im Amazon Elastic Compute Cloud-Benutzerhandbuch.

- `CAPACITY_BLOCK`— Verwenden Sie einen vorhandenen Amazon EC2 EC2-Kapazitätsblock für die ML-Reservierung. Weitere Informationen finden Sie unter Verwenden von Amazon EC2 EC2-Kapazitätsblöcken für ML mit AWS PCS.
- `INTERRUPTIBLE_CAPACITY_RESERVATION`— Verwenden Sie ein gemeinsam genutztes unterbrechbares ODCR (). I-ODCR Weitere Informationen finden Sie unter Verwenden I-ODCRs mit AWS STK..

- Es ist möglich, Slurm Konfigurationsoptionen für die Knoten in der Knotengruppe mithilfe von bereitzustellen. `--slurm-configuration` Sie können die Gewichtung (Scheduling-Priorität) und den tatsächlichen Arbeitsspeicher festlegen. Knoten mit niedrigerer Gewichtung haben eine höhere Priorität, und die Einheiten sind willkürlich. Weitere Informationen finden Sie in der Slurm Dokumentation unter Gewicht. Realer Speicher legt die Größe (in GB) des realen Speichers auf Knoten in der Knotengruppe fest. Er soll in Verbindung mit der `CR_CPU_Memory` Option für den Cluster in AWS PCS in Ihrer Slurm Konfiguration verwendet werden. Weitere Informationen finden Sie unter RealMemory in der Slurm-Dokumentation.

> **Important**
>
> Die Erstellung der Compute-Knotengruppe kann mehrere Minuten dauern.

Sie können den Status Ihrer Knotengruppe mit dem folgenden Befehl abfragen. Sie können die Knotengruppe erst dann einer Warteschlange zuordnen, wenn ihr Status erreicht ist`ACTIVE`.

```
aws pcs get-compute-node-group --region region \
    --cluster-identifier my-cluster \
    --compute-node-group-identifier my-node-group
```

Aktualisierung eines AWS PCS-Compute-Knotengruppe

Dieses Thema bietet einen Überblick über die verfügbaren Optionen und beschreibt, was bei der Aktualisierung einer AWS PCS-Rechenknotengruppe zu beachten ist. Informationen zu den benutzerdefinierten Slurm-Einstellungen finden Sie unterBenutzerdefinierte Slurm-Einstellungen für AWS PCS-Compute-Knotengruppen.

Optionen für die Aktualisierung einer AWS PCS-Rechenknotengruppe

Durch die Aktualisierung einer AWS PCS-Rechenknotengruppe können Sie die Eigenschaften der von AWS PCS gestarteten Instances sowie die Regeln für den Start dieser Instances ändern. Sie können beispielsweise das AMI für Knotengruppen-Instances durch ein anderes ersetzen, auf dem eine andere Software installiert ist. Oder Sie können Sicherheitsgruppen aktualisieren, um die eingehende oder ausgehende Netzwerkkonnektivität zu ändern. Sie können auch die Skalierungskonfiguration und die bevorzugte Kaufoption ändern.

Die folgenden Knotengruppeneinstellungen können nach der Erstellung nicht geändert werden:

- Name
- Instances

Überlegungen bei der Aktualisierung eines AWS PCS-Compute-Knotengruppe

Compute-Knotengruppen definieren EC2-Instances, die für die Verarbeitung von Jobs, für den interaktiven Shell-Zugriff und für andere Aufgaben verwendet werden. Sie sind häufig mit einer oder mehreren AWS PCS-Warteschlangen verknüpft. Wenn Sie Ihre Compute-Knotengruppe aktualisieren, um ihr Verhalten (oder das ihrer Knoten) zu ändern, sollten Sie Folgendes berücksichtigen:

- Änderungen an den Eigenschaften der Compute-Knotengruppe werden wirksam, wenn sich der Status der Compute-Knotengruppe von Aktuell auf Aktiv ändert. Neue Instances werden mit den aktualisierten Eigenschaften gestartet.
- Updates, die sich nicht auf die Konfiguration bestimmter Knoten auswirken, wirken sich nicht auf laufende Knoten aus. Zum Beispiel das Hinzufügen eines Subnetzes und das Ändern der Zuweisungsstrategie.
- Wenn Sie die Startvorlage für eine Compute-Knotengruppe aktualisieren, müssen Sie die Compute-Knotengruppe aktualisieren, um die neue Version verwenden zu können.
- Um eine Sicherheitsgruppe zu Knoten in einer Compute-Knotengruppe hinzuzufügen oder zu entfernen, bearbeiten Sie deren Startvorlage und aktualisieren Sie die Compute-Knotengruppe. Neue Instances werden mit den aktualisierten Sicherheitsgruppen gestartet.
- Wenn Sie eine Sicherheitsgruppe, die von einer Compute-Knotengruppe verwendet wird, direkt bearbeiten, wirkt sich dies sofort auf laufende und future Instances aus.

- Wenn Sie dem von einer Compute-Knotengruppe verwendeten IAM-Instanzprofil Berechtigungen hinzufügen oder daraus entfernen, wirkt sich dies sofort auf laufende und future Instances aus.
- Um das von den Instances einer Compute-Knotengruppe verwendete AMI zu ändern, aktualisieren Sie die Compute-Knotengruppe (oder ihre Startvorlage), sodass sie das neue AMI verwendet, und warten Sie, bis AWS PCS die Instances ersetzt.
- AWS PCS ersetzt bestehende Instances in der Knotengruppe nach einem Aktualisierungsvorgang für die Knotengruppe. Wenn auf einem Knoten Jobs ausgeführt werden, können diese Jobs abgeschlossen werden, bevor AWS PCS den Knoten ersetzt. Interaktive Benutzerprozesse (z. B. auf Anmeldeknoteninstanzen) werden beendet. Der Status der Knotengruppe kehrt zu dem `Active` Zeitpunkt zurück, zu dem AWS PCS die Instances als Ersatz markiert, der tatsächliche Austausch erfolgt jedoch, wenn sich die Instances im Leerlauf befinden.
- Wenn Sie die maximal zulässige Anzahl von Instanzen in einer Compute-Knotengruppe verringern, entfernt AWS PCS Knoten aus Slurm, um das neue Maximum zu erreichen. AWS PCS beendet laufende Instances, die den entfernten Slurm-Knoten zugeordnet sind. Die laufenden Jobs auf den entfernten Knoten schlagen fehl und kehren in ihre Warteschlangen zurück.
- AWS PCS erstellt für jede Compute-Knotengruppe eine verwaltete Startvorlage. Sie sind benanntpcs-*identifier*-do-not-delete. Wählen Sie sie nicht aus, wenn Sie eine Compute-Knotengruppe erstellen oder aktualisieren, da die Knotengruppe sonst nicht richtig funktioniert.
- Wenn Sie eine Compute-Knotengruppe so aktualisieren, dass sie Spot als Kaufoption verwendet, muss die AWSServiceRoleForEC2Spotserviceverknüpfte Rolle in Ihrem Konto vorhanden sein. Weitere Informationen finden Sie unter Amazon EC2 Spot-Rolle für AWS STK..

So aktualisieren Sie eine AWS PCS-Rechenknotengruppe

Sie können eine Knotengruppe mithilfe der AWS-Managementkonsole oder der AWS-CLI aktualisieren.

AWS-Managementkonsole

Um eine Compute-Knotengruppe zu aktualisieren

1. Öffnen Sie die AWS-PCS-Konsole unter `https://console.aws.amazon.com/pcs/home#/clusters`
2. Wählen Sie den Cluster aus, in dem Sie eine Rechenknotengruppe aktualisieren möchten.
3. Navigieren Sie zu Compute-Knotengruppen, gehen Sie zu der Knotengruppe, die Sie aktualisieren möchten, und wählen Sie dann Bearbeiten aus.

4. Aktualisieren Sie in den Abschnitten Computerkonfiguration, Zusätzliche Einstellungen und SlurmAnpassungseinstellungen alle Werte mit Ausnahme von:

   • Instanzen — Sie können die Instanzen in einer Compute-Knotengruppe nicht ändern.

   Weitere Informationen zu den benutzerdefinierten Slurm-Einstellungen finden Sie unterBenutzerdefinierte Slurm-Einstellungen für AWS PCS-Compute-Knotengruppen.

5. Im Abschnitt „Scheduler-Konfiguration“ können Sie die Scale-down Leerlaufzeit aktualisieren, um einen neuen Wert (1—10000000 Sekunden) festzulegen, der den Cluster-Standard überschreibt, oder den Wert löschen, um zur Einstellung auf Cluster-Ebene zurückzukehren. Für diese Einstellung ist Version 25.11 oder höher erforderlich. Slurm
6. Wählen Sie Aktualisieren aus. Im Feld Status wird die Meldung Aktualisierung angezeigt, während die Änderungen übernommen werden.

> **Important**
>
> Aktualisierungen von Compute-Knotengruppen können mehrere Minuten dauern.

AWS CLI

Um eine Compute-Knotengruppe zu aktualisieren

1. Aktualisieren Sie Ihre Compute-Knotengruppe mit dem folgenden Befehl. Nehmen Sie vor der Ausführung des Befehls die folgenden Ersetzungen vor:

   a. *region-code*Ersetzen Sie es durch die AWS-Region, in der Sie Ihren Cluster erstellen möchten.

   b. *my-node-group*Ersetzen Sie es durch den Namen oder `computeNodeGroupId` für Ihre Rechenknotengruppe.

   c. *my-cluster*Ersetzen Sie durch den Namen oder `clusterId` Ihres Clusters.

```
aws pcs update-compute-node-group --region region-code \
    --cluster-identifier my-cluster \
    --compute-node-group-identifier my-node-group
```

Example— Aktualisierung einer Compute-Knotengruppe mit benutzerdefinierten Slurm-Einstellungen

```
aws pcs update-compute-node-group --region region-code \
    --cluster-identifier my-cluster \
    --compute-node-group-identifier my-node-group \
    --slurm-configuration \
    'slurmCustomSettings=[{parameterName=Features,parameterValue="gpu,nvme"}]'
```

Weitere Informationen finden Sie unter Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Compute-Knotengruppen.

Example— Aktualisierung der Leerlaufzeit beim Herunterskalieren für eine Rechenknotengruppe

```
aws pcs update-compute-node-group --region region \
    --cluster-identifier my-cluster \
    --compute-node-group-identifier my-gpu-nodes \
    --slurm-configuration scaleDownIdleTimeInSeconds=600
```

Sie können eine Aktualisierung `scaleDownIdleTimeInSeconds` für eine bestehende Compute-Knotengruppe vornehmen, um die Leerlaufzeit beim Herunterfahren auf Clusterebene zu überschreiben. Gültige Werte sind 1—10000000. Für diese Einstellung ist Slurm Version 25.11 oder höher erforderlich.

Example— Die Überschreibung der Leerlaufzeit nach unten wird aus einer Compute-Knotengruppe entfernt

```
aws pcs update-compute-node-group --region region \
    --cluster-identifier my-cluster \
    --compute-node-group-identifier my-gpu-nodes \
    --slurm-configuration scaleDownIdleTimeInSeconds=-1
```

Auf setzen`scaleDownIdleTimeInSeconds`, `-1` um die Überschreibung der Knotengruppe zu entfernen und zur Einstellung auf Clusterebene zurückzukehren.

2. Aktualisieren Sie alle Knotengruppenparameter mit Ausnahme von. `--instance-configs` Um beispielsweise eine neue AMI-ID festzulegen, übergeben Sie `--amiId my-custom-ami-id` where *`my-custom-ami-id`* wird durch das AMI Ihrer Wahl ersetzt.

> **Important**
>
> Die Aktualisierung der Compute-Knotengruppe kann mehrere Minuten dauern.

Sie können den Status Ihrer Knotengruppe mit dem folgenden Befehl abfragen.

```
aws pcs get-compute-node-group --region region-code \
    --cluster-identifier my-cluster \
    --compute-node-group-identifier my-node-group
```

Löschen einer Compute-Knotengruppe in AWS PCS

Dieses Thema bietet einen Überblick über die verfügbaren Optionen und beschreibt, was zu beachten ist, wenn Sie eine Rechenknotengruppe in AWS PCS löschen.

Überlegungen beim Löschen einer Compute-Knotengruppe

Compute-Knotengruppen definieren EC2-Instances, die für die Verarbeitung von Jobs, für den interaktiven Shell-Zugriff und für andere Aufgaben verwendet werden. Sie sind häufig mit einer oder mehreren AWS PCS-Warteschlangen verknüpft. Bevor Sie eine Compute-Knotengruppe löschen, sollten Sie Folgendes beachten:

- Alle von der Compute-Knotengruppe gestarteten EC2-Instances werden beendet. Dadurch werden Jobs storniert, die auf diesen Instances ausgeführt werden, und laufende interaktive Prozesse werden beendet.
- Sie müssen die Zuordnung der Compute-Knotengruppe zu allen Warteschlangen aufheben, bevor Sie sie löschen können. Weitere Informationen finden Sie unter Aktualisierung einer AWS PCS-Warteschlange.

Löschen Sie die Compute-Knotengruppe

Sie können das AWS-Managementkonsole oder verwenden AWS CLI , um eine Compute-Knotengruppe zu löschen.

AWS-Managementkonsole

Um eine Compute-Knotengruppe zu löschen

1. Öffnen Sie die AWS PCS-Konsole.
2. Wählen Sie den Cluster der Compute-Knotengruppe aus.
3. Navigieren Sie zu Compute-Knotengruppen und wählen Sie die zu löschende Compute-Knotengruppe aus.
4. Wählen Sie Löschen aus.
5. Das Feld Status wird angezeigt`Deleting`. Das kann mehrere Minuten dauern.

> **Note**
>
> Sie können Befehle verwenden, die Ihrem Scheduler eigen sind, um zu bestätigen, dass die Compute-Knotengruppe gelöscht wurde. Verwenden Sie zum Beispiel `sinfo` or `squeue` für Slurm.

AWS CLI

Um eine Compute-Knotengruppe zu löschen

- Verwenden Sie den folgenden Befehl, um eine Compute-Knotengruppe mit diesen Ersetzungen zu löschen:
  - Ersetzen Sie *`region-code`* durch den, in dem sich AWS-Region Ihr Cluster befindet.
  - *`my-node-group`*Ersetzen Sie durch den Namen oder die ID Ihrer Compute-Knotengruppe.
  - *`my-cluster`*Ersetzen Sie durch den Namen oder die ID Ihres Clusters.

```
aws pcs delete-compute-node-group --region region-code \
      --compute-node-group-identifier my-node-group \
      --cluster-identifier my-cluster
```

Das Löschen der Compute-Knotengruppe kann mehrere Minuten dauern.

> **Note**
>
> Sie können Befehle verwenden, die Ihrem Scheduler eigen sind, um zu bestätigen, dass die Compute-Knotengruppe gelöscht wurde. Verwenden Sie zum Beispiel `sinfo` or `squeue` für Slurm.

Details zur Compute-Knotengruppe in AWS PCS abrufen

Sie können das AWS-Managementkonsole oder verwenden, AWS CLI um Details zu einer Rechenknotengruppe abzurufen, z. B. ihre Compute-Knotengruppen-ID, den Amazon-Ressourcennamen (ARN) und die Amazon Machine Image (AMI) -ID. Bei diesen Details handelt es sich häufig um erforderliche Werte für AWS PCS-API-Aktionen und -Konfigurationen.

AWS-Managementkonsole

Um Details zur Compute-Knotengruppe abzurufen

1. Öffnen Sie die AWS PCS-Konsole.
2. Wählen Sie den -Cluster.
3. Wählen Sie Compute Node Groups aus.
4. Wählen Sie im Listenbereich eine Compute-Knotengruppe aus.

AWS CLI

Um Details zur Compute-Knotengruppe abzurufen

1. Verwenden Sie die ListClustersAPI-Aktion, um Ihren Clusternamen oder Ihre Cluster-ID zu ermitteln.

```
aws pcs list-clusters
```

Beispielausgabe:

```
{
    "clusters": [
        {
```

```
            "name": "get-started-cfn",
            "id": "pcs_abc1234567",
            "arn": "arn:aws:pcs:us-east-1:111122223333:cluster/pcs_abc1234567",
            "createdAt": "2025-04-01T20:11:22+00:00",
            "modifiedAt": "2025-04-01T20:11:22+00:00",
            "status": "ACTIVE"
        }
    ]
}
```

2. Verwenden Sie die ListComputeNodeGroupsAPI-Aktion, um die Compute-Knotengruppen in einem Cluster aufzulisten.

```
aws pcs list-compute-node-groups --cluster-identifier cluster-name-or-id
```

Beispiel für einen Aufruf:

```
aws pcs list-compute-node-groups --cluster-identifier get-started-cfn
```

Beispielausgabe:

```
{
    "computeNodeGroups": [
        {
            "name": "compute-1",
            "id": "pcs_abc123abc1",
            "arn": "arn:aws:pcs:us-east-1:111122223333:cluster/pcs_abc1234567/
computenodegroup/pcs_abc123abc1",
            "clusterId": "pcs_abc1234567",
            "createdAt": "2025-04-01T20:19:25+00:00",
            "modifiedAt": "2025-04-01T20:19:25+00:00",
            "status": "ACTIVE"
        },
        {
            "name": "login",
            "id": "pcs_abc456abc7",
            "arn": "arn:aws:pcs:us-east-1:111122223333:cluster/pcs_abc1234567/
computenodegroup/pcs_abc456abc7",
            "clusterId": "pcs_abc1234567",
            "createdAt": "2025-04-01T20:19:31+00:00",
            "modifiedAt": "2025-04-01T20:19:31+00:00",
            "status": "ACTIVE"
```

```
        }
    ]
}
```

3. Verwenden Sie die GetComputeNodeGroupAPI-Aktion, um zusätzliche Details für eine Compute-Knotengruppe abzurufen.

```
aws pcs get-compute-node-group --cluster-identifier cluster-name-or-id --
compute-node-group-identifier compute-node-group-name-or-id
```

Beispiel für einen Aufruf:

```
aws pcs get-compute-node-group --cluster-identifier get-started-cfn --compute-
node-group-identifier compute-1
```

Beispielausgabe:

```
{
    "computeNodeGroup": {
        "name": "compute-1",
        "id": "pcs_abc123abc1",
        "arn": "arn:aws:pcs:us-east-1:111122223333:cluster/pcs_abc1234567/
computenodegroup/pcs_abc123abc1",
        "clusterId": "pcs_abc1234567",
        "createdAt": "2025-04-01T20:19:25+00:00",
        "modifiedAt": "2025-04-01T20:19:25+00:00",
        "status": "ACTIVE",
        "amiId": "ami-0123456789abcdef0",
        "subnetIds": [
            "subnet-abc012345789abc12"
        ],
        "purchaseOption": "ONDEMAND",
        "customLaunchTemplate": {
            "id": "lt-012345abcdef01234",
            "version": "1"
        },
        "iamInstanceProfileArn": "arn:aws:iam::111122223333:instance-profile/
AWSPCS-get-started-cfn-us-east-1",
        "scalingConfiguration": {
            "minInstanceCount": 0,
            "maxInstanceCount": 4
        },
```

```
        "instanceConfigs": [
            {
                "instanceType": "c6i.xlarge"
            }
        ]
    }
}
```

Suchen nach Rechenknotengruppeninstanzen in AWS PCS

Jede AWS PCS-Compute-Knotengruppe kann EC2-Instances mit gemeinsam genutzten Konfigurationen starten. Sie können EC2-Tags verwenden, um Instances in einer Compute-Knotengruppe im AWS-Managementkonsole oder mit dem zu finden. AWS CLI

AWS-Managementkonsole

Um Ihre Compute-Knotengruppen-Instances zu finden

1. Öffnen Sie die AWS PCS-Konsole.
2. Wählen Sie den -Cluster.
3. Wählen Sie Compute Node Groups aus.
4. Suchen Sie die ID für die Login-Knotengruppe, die Sie erstellt haben.
5. Navigieren Sie zur EC2-Konsole und wählen Sie Instances aus.
6. Suchen Sie nach den Instances mit dem folgenden Tag. *node-group-id*Ersetzen Sie es durch die ID (nicht den Namen) Ihrer Rechenknotengruppe.

   ```
   aws:pcs:compute-node-group-id=node-group-id
   ```
7. (Optional) Sie können den Wert von Instance state im Suchfeld ändern, um nach Instances zu suchen, die gerade konfiguriert werden oder die kürzlich beendet wurden.
8. Suchen Sie die Instanz-ID und IP-Adresse für jede Instanz in der Liste der markierten Instanzen.

AWS CLI

Verwenden Sie die folgenden Befehle, um Ihre Knotengruppen-Instances zu finden. Nehmen Sie vor dem Ausführen der Befehle die folgenden Ersetzungen vor:

- *region-code*Ersetzen Sie es durch das AWS-Region Ihres Clusters. Beispiel: `us-east-1`
- *node-group-id*Ersetzen Sie durch die ID (nicht den Namen) Ihrer Rechenknotengruppe. Informationen zur ID einer Compute-Knotengruppe finden Sie unterDetails zur Compute-Knotengruppe in AWS PCS abrufen.
- `running`Ersetzen Sie diese durch andere Instanzstatus, z. B. durch `pending` oder`terminated`, um EC2-Instances in anderen Status zu finden.

```
aws ec2 describe-instances \
     --region region-code --filters \
     "Name=tag:aws:pcs:compute-node-group-id,Values=node-group-id" \
     "Name=instance-state-name,Values=running" \
     --query 'Reservations[*].Instances[*].
{InstanceID:InstanceId,State:State.Name,PublicIP:PublicIpAddress,PrivateIP:PrivateIpAddress}
```

Daraufhin erhalten Sie ein Ergebnis, das dem hier dargestellten entspricht. Der Wert von `PublicIP` ist, `null` wenn sich die Instance in einem privaten Subnetz befindet.

```
[
    [
        {
            "InstanceID": "i-0123456789abcdefa",
            "State": "running",
            "PublicIP": "18.189.32.188",
            "PrivateIP": "10.0.0.1"
        }
    ]
]
```

> **Note**
>
> Wenn Sie damit `describe-instances` rechnen, eine große Anzahl von Instances zurückzugeben, müssen Sie Optionen für mehrere Seiten verwenden. Weitere Informationen finden Sie DescribeInstancesin der Amazon Elastic Compute Cloud API-Referenz.

Verwenden von Amazon EC2 EC2-Startvorlagen mit AWS PCS

In Amazon EC2 kann eine Startvorlage eine Reihe von Einstellungen speichern, sodass Sie diese beim Starten von Instances nicht einzeln angeben müssen. AWS PCS enthält Startvorlagen als flexible Methode zur Konfiguration von Rechenknotengruppen. Wenn Sie eine Knotengruppe erstellen, stellen Sie eine Startvorlage bereit. AWS PCS erstellt daraus eine abgeleitete Startvorlage, die Transformationen enthält, um sicherzustellen, dass sie mit dem Service funktioniert.

Wenn Sie wissen, welche Optionen und Überlegungen beim Schreiben einer benutzerdefinierten Startvorlage zu beachten sind, können Sie eine Vorlage für die Verwendung mit AWS PCS erstellen. Weitere Informationen zu Startvorlagen finden Sie unter Launching an Instance from a Launch an Instance from a Launch Template im Amazon EC2 EC2-Benutzerhandbuch.

Themen

Überblick über Startvorlagen in AWS PCS

Es stehen über 30 Parameter zur Verfügung, die Sie in eine EC2-Startvorlage aufnehmen können und die viele Aspekte der Konfiguration von Instances steuern. Die meisten sind vollständig mit AWS PCS kompatibel, es gibt jedoch einige Ausnahmen.

Die folgenden Parameter der EC2 Launch-Vorlage werden von AWS PCS ignoriert, da diese Eigenschaften direkt vom Dienst verwaltet werden müssen:

- Attribute des type/Specify Instance-Instance-Typs (`InstanceRequirements`) — AWS PCS unterstützt keine attributbasierte Instance-Auswahl.
- Instanztyp (`InstanceType`) — Geben Sie Instanztypen an, wenn Sie eine Knotengruppe erstellen.

- Erweitertes details/IAM Instanzprofil (`IamInstanceProfile`) — Dieses geben Sie an, wenn Sie die Knotengruppe erstellen oder aktualisieren.
- Erweiterte details/Disable API-Terminierung (`DisableApiTermination`) — AWS PCS muss den Lebenszyklus der von ihm gestarteten Knotengruppen-Instances kontrollieren.
- Erweiterter details/Disable API-Stopp (`DisableApiStop`) — AWS PCS muss den Lebenszyklus der von ihm gestarteten Knotengruppen-Instances kontrollieren.
- Erweitert details/Stop — Verhalten im Ruhezustand (`HibernationOptions`) — AWS PCS unterstützt den Ruhezustand von Instanzen nicht.
- Advanced details/Elastic GPU (`ElasticGpuSpecifications`) — Amazon Elastic Graphics hat am 8. Januar 2024 das Ende der Nutzungsdauer erreicht.
- Erweiterte details/Elastic Inferenz (`ElasticInferenceAccelerators`) — Amazon Elastic Inference ist für Neukunden nicht mehr verfügbar.
- AAdvanced details/Specify CPU options/Threadspro Kern (`ThreadsPerCore`) — AWS PCS legt die Anzahl der Threads pro Kern auf 1 fest.

Für diese Parameter gelten spezielle Anforderungen, die die Kompatibilität mit AWS PCS unterstützen:

- Benutzerdaten (`UserData`) — Diese müssen mehrteilig codiert sein. Siehe Arbeiten mit Amazon EC2 EC2-Benutzerdaten für AWS PCS.
- Anwendungs- und Betriebssystem-Images (`ImageId`) — Sie können dies einschließen. Wenn Sie jedoch beim Erstellen oder Aktualisieren der Knotengruppe eine AMI-ID angeben, überschreibt diese den Wert in der Startvorlage. Das von Ihnen bereitgestellte AMI muss mit AWS PCS kompatibel sein. Weitere Informationen finden Sie unter "Amazon Machine Images (AMIs) für AWS STK..
- Netzwerk settings/Firewall (Sicherheitsgruppen) (`SecurityGroups`) — Eine Liste von Sicherheitsgruppennamen kann in einer AWS PCS-Startvorlage nicht festgelegt werden. Sie können eine Liste von Sicherheitsgruppen IDs (`SecurityGroupIds`) einrichten, es sei denn, Sie definieren Netzwerkschnittstellen in der Startvorlage. Anschließend müssen Sie IDs für jede Schnittstelle eine Sicherheitsgruppe angeben. Weitere Informationen finden Sie unter Sicherheitsgruppen in AWS PCS.
- settings/Advanced Netzwerkkonfiguration (`NetworkInterfaces`) — Wenn Sie EC2-Instances mit einer einzigen Netzwerkkarte verwenden und keine spezielle Netzwerkkonfiguration benötigen, kann AWS PCS das Instance-Netzwerk für Sie konfigurieren. Um mehrere Netzwerkkarten zu konfigurieren oder den Elastic Fabric Adapter auf Ihren Instances zu aktivieren, verwenden

SieNetworkInterfaces. IDs Unter jeder Netzwerkschnittstelle muss eine Liste der Sicherheitsgruppen enthalten seinGroups. Weitere Informationen finden Sie unter Mehrere Netzwerkschnittstellen in AWS PCS.

- Erweiterte Details/Kapazitätsreservierung (CapacityReservationSpecification) — Dies kann eingestellt werden, kann aber CapacityReservationId bei der Arbeit mit AWS PCS nicht auf ein bestimmtes Objekt verweisen. Sie können jedoch auf eine Kapazitätsreservierungsgruppe verweisen, wenn diese Gruppe eine oder mehrere Kapazitätsreservierungen enthält. Weitere Informationen finden Sie unter Kapazitätsreservierungen in AWS STK..

Erstellen einer grundlegenden Startvorlage

Sie können eine Startvorlage mit dem AWS-Managementkonsole oder dem erstellen AWS CLI.

AWS-Managementkonsole

Eine Startvorlage erstellen

1. Öffnen Sie die EC2Amazon-Konsole und wählen Sie Vorlagen starten aus.
2. Wählen Sie Startvorlage erstellen.
3. Geben Sie unter Name und Beschreibung der Startvorlage einen eindeutigen, unverwechselbaren Namen für den Namen der Startvorlage ein
4. Wählen Sie unter key pair (Anmeldung) bei Schlüsselpaarname das SSH-Schlüsselpaar aus, das für die Anmeldung bei von AWS PCS verwalteten EC2 Instanzen verwendet werden soll. Dies ist zwar optional, wird aber empfohlen.
5. Wählen Sie unter Netzwerkeinstellungen und dann Firewall (Sicherheitsgruppen) die Sicherheitsgruppen aus, die an die Netzwerkschnittstelle angehängt werden sollen. Alle Sicherheitsgruppen in der Startvorlage müssen aus Ihrer AWS PCS-Cluster-VPC stammen. Wählen Sie mindestens:

   - Eine Sicherheitsgruppe, die die Kommunikation mit dem AWS PCS-Cluster ermöglicht
   - Eine Sicherheitsgruppe, die die Kommunikation zwischen EC2 Instances ermöglicht, die von AWS PCS gestartet wurden
   - (Optional) Eine Sicherheitsgruppe, die eingehenden SSH-Zugriff auf interaktive Instanzen ermöglicht
   - (Optional) Eine Sicherheitsgruppe, die es Rechenknoten ermöglicht, ausgehende Verbindungen zum Internet herzustellen

- (Optional) Sicherheitsgruppe (n), die den Zugriff auf Netzwerkressourcen wie gemeinsam genutzte Dateisysteme oder einen Datenbankserver ermöglichen.

6. Ihre neue Startvorlagen-ID ist in der EC2 Amazon-Konsole unter Startvorlagen verfügbar. Die ID der Startvorlage wird das folgende Formular haben`lt-0123456789abcdef01`.

Als nächster Schritt wird empfohlen

- Verwenden Sie die neue Startvorlage, um eine AWS PCS-Compute-Knotengruppe zu erstellen oder zu aktualisieren.

AWS CLI

Eine Startvorlage erstellen

Erstellen Sie Ihre Startvorlage mit dem folgenden Befehl.

- Nehmen Sie vor der Ausführung des Befehls die folgenden Ersetzungen vor:

  a. *region-code*Ersetzen Sie es durch die AWS-Region Stelle, an der Sie mit AWS PCS arbeiten

  b. *my-launch-template-name*Ersetzen Sie es durch einen Namen für Ihre Vorlage. Es muss für das AWS-Konto und, das AWS-Region Sie verwenden, eindeutig sein.

  c. *my-ssh-key-name*Ersetzen Sie es durch den Namen Ihres bevorzugten SSH-Schlüssels.

  d. Ersetzen Sie *sg-ExampleID1* und *sg-ExampleID2* durch eine Sicherheitsgruppe IDs , die die Kommunikation zwischen Ihren EC2 Instances und dem Scheduler sowie die Kommunikation zwischen EC2 Instanzen ermöglicht. Wenn Sie nur über eine Sicherheitsgruppe verfügen, die den gesamten Datenverkehr ermöglicht, können Sie das vorangegangene Kommazeichen entfernen`sg-ExampleID2`. Sie können auch weitere Sicherheitsgruppen IDs hinzufügen. Alle Sicherheitsgruppen, die Sie in die Startvorlage aufnehmen, müssen aus Ihrer AWS PCS-Cluster-VPC stammen.

```
aws ec2 create-launch-template --region region-code \
    --launch-template-name my-template-name \
    --launch-template-data '{"KeyName":"my-ssh-key-name","SecurityGroupIds":
 ["sg-ExampleID1","sg-ExampleID2"]}'
```

Es AWS CLI wird Text ausgegeben, der dem folgenden ähnelt. Die ID der Startvorlage befindet sich in. `LaunchTemplateId`

```
{
    "LaunchTemplate": {
        "LatestVersionNumber": 1,
        "LaunchTemplateId": "lt-0123456789abcdef01",
        "LaunchTemplateName": "my-launch-template-name",
        "DefaultVersionNumber": 1,
        "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
        "CreateTime": "2019-04-30T18:16:06.000Z"
    }
}
```

Als nächster Schritt wird empfohlen

- Verwenden Sie die neue Startvorlage, um eine AWS PCS-Compute-Knotengruppe zu erstellen oder zu aktualisieren.

Arbeiten mit Amazon EC2 EC2-Benutzerdaten für AWS PCS

Sie können EC2-Benutzerdaten in Ihrer Startvorlage angeben, die beim Start Ihrer Instances `cloud-init` ausgeführt wird. Benutzerdatenblöcke mit dem Inhaltstyp werden `cloud-config` ausgeführt, bevor sich die Instance bei der AWS PCS-API registriert, während Benutzerdatenblöcke mit dem Inhaltstyp nach Abschluss der Registrierung, aber bevor der Slurm-Daemon gestartet wird, `text/x-shellscript` ausgeführt werden. Weitere Informationen zu Inhaltstypen finden Sie in der Cloud-Init-Dokumentation.

Mit unseren Benutzerdaten können gängige Konfigurationsszenarien durchgeführt werden, einschließlich, aber nicht beschränkt auf die folgenden:

- Einschließlich Benutzer oder Gruppen
- Pakete werden installiert
- Partitionen und Dateisysteme erstellen
- Mounten von Netzwerk-Dateisystemen

Benutzerdaten in Startvorlagen müssen im mehrteiligen MIME-Archivformat vorliegen. Dies liegt daran, dass Ihre Benutzerdaten mit anderen AWS PCS-Benutzerdaten zusammengeführt werden,

die für die Konfiguration von Knoten in Ihrer Knotengruppe erforderlich sind. Sie können mehrere Benutzerdatenblöcke in einer einzelnen mehrteiligen MIME-Datei kombinieren.

Eine mehrteilige MIME-Datei umfasst folgende Komponenten:

- Deklaration von Inhaltstyp und Teilgrenze: `Content-Type: multipart/mixed; boundary="==BOUNDARY=="`
- Deklaration der MIME-Version: `MIME-Version: 1.0`
- Ein oder mehrere Benutzerdatenblöcke, die die folgenden Komponenten enthalten:
  - Die Öffnungsgrenze, die den Beginn eines Benutzerdatenblocks signalisiert:`--==BOUNDARY==`. Sie müssen die Zeile vor dieser Grenze leer lassen.
  - Die Inhaltstyp-Deklaration für den Block: `Content-Type: text/cloud-config; charset="us-ascii"` oder`Content-Type: text/x-shellscript; charset="us-ascii"`. Sie müssen die Zeile nach der Inhaltstyp-Deklaration leer lassen.
  - Der Inhalt der Benutzerdaten, z. B. eine Liste von Shell-Befehlen oder `cloud-config` -Direktiven.
- Die schließende Grenze, die das Ende der mehrteiligen MIME-Datei signalisiert:`--==BOUNDARY==--`. Sie müssen die Zeile vor der schließenden Grenze leer lassen.

> **Note**
>
> Wenn Sie Benutzerdaten zu einer Startvorlage in der Amazon EC2 EC2-Konsole hinzufügen, können Sie sie als Klartext einfügen. Oder Sie können es aus einer Datei hochladen. Wenn Sie das AWS CLI oder ein AWS SDK verwenden, müssen Sie zuerst die Benutzerdaten base64-kodieren und diese Zeichenfolge beim Aufrufen als Wert des `UserData` Parameters angeben CreateLaunchTemplate, wie in dieser JSON-Datei gezeigt.

```
{
    "LaunchTemplateName": "base64-user-data",
    "LaunchTemplateData": {
        "UserData":
 "ewogICAgIkxhdW5jaFRlbXBsYXRlTmFtZSI6ICJpbmNyZWFzZS1jb250YWluZXItdm9sdW..."
    }
}
```

Beispiele

- Beispiel: Software aus einem Paket-Repository installieren
- Beispiel: Führen Sie Skripts aus einem S3-Bucket aus
- Beispiel: Legen Sie globale Umgebungsvariablen fest
- Verwenden von Netzwerkdateisystemen mit AWS PCS
- Beispiel: Verwenden Sie ein EFS-Dateisystem als gemeinsam genutztes Home-Verzeichnis

Beispiel: Software für AWS PCS aus einem Paket-Repository installieren

Geben Sie dieses Skript als Wert von `"userData"` in Ihrer Startvorlage an. Weitere Informationen finden Sie unter Arbeiten mit Amazon EC2 EC2-Benutzerdaten für AWS PCS.

Dieses Skript verwendet cloud-config, um beim Start Softwarepakete auf Knotengruppen-Instances zu installieren. Weitere Informationen finden Sie unter Benutzerdatenformate in der Cloud-Init-Dokumentation. In diesem Beispiel wird und installiert`curl`. `llvm`

> **Note**
>
> Ihre Instances müssen in der Lage sein, eine Verbindung zu ihren konfigurierten Paket-Repositorys herzustellen.

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="==MYBOUNDARY=="

--==MYBOUNDARY==
Content-Type: text/cloud-config; charset="us-ascii"

packages:
- python3-devel
- rust
- golang

--==MYBOUNDARY==--
```

Beispiel: Zusätzliche Skripts für AWS PCS aus einem S3-Bucket ausführen

Geben Sie dieses Skript als Wert von `"userData"` in Ihrer Startvorlage an. Weitere Informationen finden Sie unter Arbeiten mit Amazon EC2 EC2-Benutzerdaten für AWS PCS.

Das folgende Benutzerdatenskript verwendet cloud-config, um ein Skript aus einem S3-Bucket zu importieren und es beim Start auf Knotengruppen-Instances auszuführen. Weitere Informationen finden Sie unter Benutzerdatenformate in der Cloud-Init-Dokumentation.

Ersetzen Sie die folgenden Werte durch Ihre eigenen Daten:

- *amzn-s3-demo-bucket*— Der Name eines S3-Buckets, aus dem Ihr Konto lesen kann.
- *object-key*— Der S3-Objektschlüssel des zu importierenden Skripts. Dazu gehören der Name des Skripts und sein Speicherort in der Ordnerstruktur des Buckets. Beispiel, `scripts/script.sh`. Weitere Informationen finden Sie unter Organisieren von Objekten in der Amazon S3 S3-Konsole mithilfe von Ordnern im Amazon Simple Storage Service-Benutzerhandbuch.
- *shell*— Die Linux-Shell, die zur Ausführung des Skripts verwendet werden soll, z. `bash` B.

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="==MYBOUNDARY=="

--==MYBOUNDARY==
Content-Type: text/cloud-config; charset="us-ascii"

runcmd:
- aws s3 cp s3://amzn-s3-demo-bucket/object-key /tmp/script.sh
- /usr/bin/shell /tmp/script.sh

--==MYBOUNDARY==--
```

Das IAM-Instanzprofil für die Knotengruppe muss Zugriff auf den Bucket haben. Die folgende IAM-Richtlinie ist ein Beispiel für den Bucket im obigen Benutzerdatenskript.

JSON

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

```
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Beispiel: Legen Sie globale Umgebungsvariablen für AWS PCS fest

Geben Sie dieses Skript als Wert von `"userData"` in Ihrer Startvorlage an. Weitere Informationen finden Sie unter Arbeiten mit Amazon EC2 EC2-Benutzerdaten für AWS PCS.

Im folgenden Beispiel werden globale Variablen `/etc/profile.d` für Knotengruppen-Instances festgelegt.

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="==MYBOUNDARY=="

--==MYBOUNDARY==
Content-Type: text/x-shellscript; charset="us-ascii"

#!/bin/bash
touch /etc/profile.d/awspcs-userdata-vars.sh
echo MY_GLOBAL_VAR1=100 >> /etc/profile.d/awspcs-userdata-vars.sh
echo MY_GLOBAL_VAR2=abc >> /etc/profile.d/awspcs-userdata-vars.sh

--==MYBOUNDARY==--
```

Beispiel: Verwenden Sie ein EFS-Dateisystem als gemeinsam genutztes Home-Verzeichnis für AWS PCS

Geben Sie dieses Skript als Wert für `"userData"` in Ihrer Startvorlage an. Weitere Informationen finden Sie unter Arbeiten mit Amazon EC2 EC2-Benutzerdaten für AWS PCS.

In diesem Beispiel wird das EFS-Mount-In zum Beispiel erweitertVerwenden von Netzwerkdateisystemen mit AWS PCS, um ein gemeinsam genutztes Home-Verzeichnis zu implementieren. Der Inhalt von /home wird gesichert, bevor das EFS-Dateisystem bereitgestellt wird. Die Inhalte werden dann nach Abschluss des Mounts schnell an ihren Platz auf dem gemeinsam genutzten Speicher kopiert.

Ersetzen Sie die folgenden Werte in diesem Skript durch Ihre eigenen Daten:

- *`/mount-point-directory`*— Der Pfad auf einer Instanz, auf der Sie das EFS-Dateisystem mounten möchten.
- *`filesystem-id`*— Die Dateisystem-ID für das EFS-Dateisystem.

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="==MYBOUNDARY=="

--==MYBOUNDARY==
Content-Type: text/cloud-config; charset="us-ascii"

packages:
  - amazon-efs-utils

runcmd:
  - mkdir -p /tmp/home
  - rsync -a /home/ /tmp/home
  - echo "filesystem-id:/ /mount-point-directory efs tls,_netdev" >> /etc/fstab
  - mount -a -t efs defaults
  - rsync -a --ignore-existing /tmp/home/ /home
  - rm -rf /tmp/home/

--==MYBOUNDARY==--
```

Beispiel: Passwortloses SSH aktivieren

Sie können auf dem Beispiel für ein gemeinsam genutztes Home-Verzeichnis aufbauen, um SSH-Verbindungen zwischen Clusterinstanzen mithilfe von SSH-Schlüsseln zu implementieren. Führen Sie für jeden Benutzer, der das Shared Home-Dateisystem verwendet, ein Skript aus, das dem folgenden ähnelt:

```
#!/bin/bash
```

```
mkdir -p $HOME/.ssh && chmod 700 $HOME/.ssh
touch $HOME/.ssh/authorized_keys
chmod 600 $HOME/.ssh/authorized_keys

if [ ! -f "$HOME/.ssh/id_rsa" ]; then
    ssh-keygen -t rsa -b 4096 -f $HOME/.ssh/id_rsa -N ""
    cat ~/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys
fi
```

> **Note**
>
> Die Instanzen müssen eine Sicherheitsgruppe verwenden, die SSH-Verbindungen zwischen Clusterknoten ermöglicht.

Kapazitätsreservierungen in AWS STK.

Mithilfe von Kapazitätsreservierungen oder Amazon EC2 EC2-Kapazitätsblöcken für ML können Sie Amazon EC2 On-Demand EC2-Kapazität in einer bestimmten Availability Zone und für einen bestimmten Zeitraum reservieren, um sicherzustellen, dass Ihnen die erforderliche Rechenkapazität zur Verfügung steht, wenn Sie sie benötigen.

On-Demand Mit Kapazitätsreservierungen (ODCRs) können Sie Rechenkapazität für Ihre Amazon EC2 EC2-Instances in einer bestimmten Availability Zone für einen beliebigen Zeitraum reservieren. Sie können Reservierungen jederzeit ohne langfristige Verpflichtungen oder Vorauszahlungen erstellen und stornieren. ODCRs sind ideal, wenn Sie flexible Kapazitätsreservierungen benötigen, die Sie an Ihre Anforderungen anpassen können. Weitere Informationen finden Sie unter On-Demand Kapazitätsreservierungen im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Mit Amazon EC2 Capacity Blocks for ML können Sie GPU-based Accelerated Computing-Instances bis zu 8 Wochen im Voraus für die future Verwendung reservieren. Sie können Blöcke mit 1 bis 64 Instances für eine Dauer von 1 Tag bis 6 Monaten reservieren. Kapazitätsblöcke eignen sich ideal für Machine-Learning-Workloads, die zu bestimmten Zeiten garantierten Zugriff auf die GPU-Kapazität erfordern. Weitere Informationen finden Sie unter Capacity Blocks for ML im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Themen

- Verwendung ODCRs mit AWS PCS
- Verwenden I-ODCRs mit AWS STK.

- Verwenden von Amazon EC2 EC2-Kapazitätsblöcken für ML mit AWS PCS

Verwendung ODCRs mit AWS PCS

Sie können wählen, wie AWS PCS Ihre Reserved Instances nutzt. Wenn Sie ein offenes ODCR erstellen, werden alle passenden Instances, die von AWS PCS oder anderen Prozessen in Ihrem Konto gestartet wurden, auf die Reservierung angerechnet. Bei einem gezielten ODCR werden nur Instances, die mit der spezifischen Reservierungs-ID gestartet wurden, auf die Reservierung angerechnet. Bei zeitkritischen Workloads ODCRs sind gezielte Workloads üblicher.

Sie können eine AWS PCS-Compute-Knotengruppe so konfigurieren, dass sie ein zielgerichtetes ODCR verwendet, indem Sie es zu einer Startvorlage hinzufügen. Gehen Sie dazu wie folgt vor:

1. Erstellen Sie mithilfe des Benutzerleitfadens Amazon EC2 Create a Capacity Reservation User Guide eine gezielte On-Demand-Kapazitätsreservierung (ODCR).
2. Ordnen Sie das ODCR einer Startvorlage zu. Es gibt zwei Möglichkeiten, dies zu tun:

    a. Direkte ODCR-Zuordnung: Verweisen Sie direkt in der Startvorlage auf die ODCR-ID. Dieser Ansatz bietet eine strikte Kapazitätskontrolle und unterstützt kein Instanz-Backfilling (wenn die Compute-Knotengruppe mehr Instanzen anfordert, als im ODCR verfügbar sind, werden keine zusätzlichen Instanzen gestartet).

    b. Zuordnung der Kapazitätsreservierungsgruppe: Fügen Sie das ODCR einer Kapazitätsreservierungsgruppe hinzu und verweisen Sie in der Startvorlage auf die Gruppe. Dieser Ansatz unterstützt das Auffüllen von Instanzen, sodass AWS PCS zusätzliche On-Demand-Instances starten kann, wenn die Reservierungskapazität überschritten wird.
3. Erstellen oder aktualisieren Sie eine AWS PCS-Compute-Knotengruppe, um die Startvorlage zu verwenden. Weitere Informationen finden Sie im AWS PCS Compute Node Groups-Benutzerhandbuch.

    - Stellen Sie den `purchaseOption` Wert der Compute-Knotengruppe auf ein`ONDEMAND`.

Beispiel: Reservieren und verwenden Sie hpc6a.48xlarge-Instances mit einem gezielten ODCR

Dieser Beispielbefehl erstellt ein Ziel-ODCR für 32 hpc6a.48xlarge-Instances. Um die Reserved Instances in einer Platzierungsgruppe zu starten, fügen Sie dem Befehl etwas hinzu. `--placement-`

`group-arn` Sie können mit `--end-date` und ein Enddatum definieren`--end-date-type`, andernfalls wird die Reservierung so lange fortgesetzt, bis sie manuell beendet wird.

```
aws ec2 create-capacity-reservation \
    --instance-type hpc6a.48xlarge \
    --instance-platform Linux/UNIX \
    --availability-zone us-east-2a \
    --instance-count 32 \
    --instance-match-criteria targeted
```

Das Ergebnis dieses Befehls ist ein ARN für das neue ODCR. Die ODCR-ID kann aus dem ARN `"arn:aws:ec2:us-east-2:123456789012:capacity-reservation/ODCR-ID"` oder mithilfe von Amazon DescribeCapacityReservations EC2 abgerufen werden.

Direkte ODCR-Zuordnung: Fügen Sie die ODCR-ID zur Startvorlage hinzu. Hier ist ein Beispiel für eine Startvorlage, die auf die ODCR-ID verweist.

```
{
  "CapacityReservationSpecification": {
    "CapacityReservationTarget": {
      "CapacityReservationId": "cr-1234567890abcdef1"
    }
  }
}
```

Zuordnung von Kapazitätsreservierungsgruppen: Erstellen Sie eine Kapazitätsreservierungsgruppe und fügen Sie die Gruppe zur Startvorlage hinzu. Mit dem folgenden Befehl wird eine Kapazitätsreservierungsgruppe mit dem Namen erstelltEXAMPLE-CR-GROUP.

```
aws resource-groups create-group \
    --name EXAMPLE-CR-GROUP \
    --configuration \
        '{"Type": "AWS::EC2::CapacityReservationPool"}' \
        '{"Type": "AWS::ResourceGroups::Generic", "Parameters": [{"Name": "allowed-
resource-types", "Values": ["AWS::EC2::CapacityReservation"]}]}'
```

Mit dem folgenden Befehl wird das ODCR zur Gruppe „Kapazitätsreservierung“ hinzugefügt.

```
aws resource-groups group-resources --group EXAMPLE-CR-GROUP \
    --resource-arns arn:aws:ec2:us-east-2:123456789012:capacity-reservation/
cr-1234567890abcdef1
```

Nachdem das ODCR erstellt und zu einer Kapazitätsreservierungsgruppe hinzugefügt wurde, kann es nun mit einer AWS PCS-Compute-Knotengruppe verbunden werden, indem es zu einer Startvorlage hinzugefügt wird. Hier ist ein Beispiel für eine Startvorlage, die auf die Kapazitätsreservierungsgruppe verweist.

```
{
  "CapacityReservationSpecification": {
    "CapacityReservationResourceGroupArn": "arn:aws:resource-groups:us-
east-2:123456789012:group/EXAMPLE-CR-GROUP"
  }
}
```

Erstellen oder aktualisieren Sie abschließend eine AWS PCS-Compute-Knotengruppe, um hpc6a.48xlarge-Instances zu verwenden, und verwenden Sie die Startvorlage, die auf die ODCR verweist. Legen Sie für eine statische Knotengruppe die Mindest- und Höchstzahl der Instanzen auf die Größe der Reservierung fest (32). Legen Sie für eine dynamische Knotengruppe die Mindestanzahl der Instanzen auf 0 und die Höchstzahl auf die gewünschte Instanzgröße fest.

Dieses Beispiel ist eine einfache Implementierung eines einzelnen ODCR, das für eine Rechenknotengruppe bereitgestellt wird. AWS PCS unterstützt jedoch viele andere Designs. Sie können beispielsweise eine große ODCR- oder Kapazitätsreservierungsgruppe auf mehrere Rechenknotengruppen aufteilen. Oder Sie können ODCRs das verwenden, das ein anderes AWS-Konto erstellt und mit Ihrem geteilt wurde.

Weitere Informationen finden Sie unter On-Demand-Kapazitätsreservierungen und Kapazitätsblöcke für ML im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Verwenden I-ODCRs mit AWS STK.

Mithilfe von Reservierungen für unterbrechbare On-Demand Kapazitäten (I-ODCRs) können ODCR-Besitzer ungenutzte reservierte Kapazität vorübergehend mit anderen Konten in ihrer AWS Organisation teilen. Consumer-Instances erhalten eine 2-minütige Kündigungswarnung, wenn der Eigentümer Kapazität zurückfordert, sodass sie für fehlertolerante Workloads wie Stapelverarbeitung, ML-Training und Datenanalyse I-ODCRs geeignet sind.

Weitere Informationen zu I-ODCRs finden Sie unter Unterbrechungskapazitätsreservierungen im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Wie I-ODCRs arbeiten Sie mit AWS STK.

Ein I-ODCR wird aus einem vorhandenen Quell-ODCR erstellt. Der Besitzer gibt an, wie viele Instanzen der unterbrechbaren Reservierung zugewiesen werden sollen. Diese Instanzen werden vom Quell-ODCR auf das neue übertragen. I-ODCR Der Eigentümer kann jederzeit Kapazität zurückfordern, wodurch Consumer-Instances mit einer Kündigungsfrist von 2 Minuten beendet werden.

Wichtigste Merkmale:

- I-ODCRs sind standardmäßig adressiert — Verbraucher müssen in ihrer Startkonfiguration auf die Reservierungs-ID verweisen.
- I-ODCRs kann nicht zu Kapazitätsreservierungsgruppen hinzugefügt werden.
- Pro Quell-ODCR kann nur eine unterbrechbare Zuordnung erstellt werden.
- Wenn der Eigentümer Kapazität zurückfordert, gibt es keinen Fallback auf On-Demand oder Spot-Consumer-Instances.

Konfiguration eines AWS PCS-Compute-Knotengruppe zur Verwendung einer I-ODCR

Sie können eine AWS PCS-Compute-Knotengruppe so konfigurieren, dass sie eine gemeinsam genutzte Knotengruppe verwendet, I-ODCR indem Sie sie zu einer Startvorlage hinzufügen. Hier sind die Schritte:

- Stellen Sie sicher, dass Sie Zugriff auf die haben I-ODCR. Der ODCR-Besitzer muss die unterbrechbare Reservierung mithilfe von AWS Resource Access Manager (RAM) mit Ihrem Konto teilen. Sobald es geteilt wurde, I-ODCR erscheint es in Ihrem Konto unter Kapazitätsreservierungen auf der Amazon EC2 EC2-Konsole.
- Erstellen Sie eine Startvorlage, die auf die I-ODCR abzielt. Verweisen Sie direkt auf die I-ODCR ID und legen Sie den Markttyp auf fest`interruptible-capacity-reservation`. Hier ist ein Beispiel für eine Startvorlage:

```
{
  "CapacityReservationSpecification": {
    "CapacityReservationTarget": {
      "CapacityReservationId": "cr-1234567890abcdef1"
    }
  },
  "InstanceMarketOptions": {
```

```
    "MarketType": "interruptible-capacity-reservation"
  }
}
```

- Erstellen oder aktualisieren Sie eine AWS PCS-Compute-Knotengruppe, um die Startvorlage zu verwenden. Weitere Informationen finden Sie unter AWS PCS-Compute-Knotengruppen.
  - Stellen Sie die `purchaseOption` der Compute-Knotengruppe auf ein`INTERRUPTIBLE_CAPACITY_RESERVATION`.

Umgang mit Unterbrechungen

Wenn der I-ODCR Eigentümer Kapazität zurückfordert, erhalten Consumer-Instances von Amazon eine zweiminütige Kündigungswarnung. EventBridge Gehen Sie wie folgt vor, um Unterbrechungen Ihrer PCS-Workloads ordnungsgemäß zu handhaben: AWS

- Konfigurieren Sie Ihre Anwendungen so, dass sie auf Unterbrechungsereignisse warten. EventBridge
- Implementieren Sie Checkpointing, damit Jobs Zwischenergebnisse speichern und nach einer Unterbrechung wieder aufgenommen werden können.
- Legen Sie für Compute-Knotengruppen mit einer dynamischen Skalierungskonfiguration die Mindestanzahl der Instanzen auf fest, `0` sodass die Gruppe problemlos herunterskaliert werden kann, wenn Kapazität zurückgewonnen wird.

Weitere Informationen zur Überwachung von Unterbrechungsereignissen finden Sie unter Überwachen von unterbrechbaren Kapazitätsreservierungen mit EventBridge im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Beispiel: Teilen und verwenden Sie hpc7a.96xlarge-Instances mit einem I-ODCR

In diesem Beispiel wird beschrieben, wie I-ODCR aus einem vorhandenen ODCR eine erstellt und mit einer PCS-Compute-Knotengruppe verwendet wird. AWS

Schritt 1: Erstellen Sie die unterbrechbare Reservierung aus einem Quell-ODCR.

Der ODCR-Besitzer erstellt aus seiner bestehenden Reservierung mit 32 Instanzen eine unterbrechbare Zuteilung von 16 Instanzen:

```
aws ec2 create-interruptible-capacity-reservation-allocation \
```

```
    --capacity-reservation-id cr-source1234567890a \
    --instance-count 16
```

Das Quell-ODCR zeigt jetzt 16 Instanzen an, und es I-ODCR wird eine neue mit 16 Instanzen erstellt.

Schritt 2: Teilen Sie das I-ODCR verwendende AWS RAM.

Der Eigentümer teilt das I-ODCR mit dem Verbraucherkonto:

```
aws ram create-resource-share \
    --name "HPC-Interruptible-Share" \
    --resource-arns arn:aws:ec2:us-east-2:123456789012:capacity-reservation/cr-
interruptible456 \
    --principals 987654321098
```

Schritt 3: Erstellen Sie eine Startvorlage für die I-ODCR.

Der Verbraucher erstellt eine Vorlage für die Markteinführung:

```
{
  "CapacityReservationSpecification": {
    "CapacityReservationTarget": {
      "CapacityReservationId": "cr-interruptible456"
    }
  },
  "InstanceMarketOptions": {
    "MarketType": "interruptible-capacity-reservation"
  }
}
```

Schritt 4: Erstellen Sie mithilfe der Startvorlage eine AWS PCS-Compute-Knotengruppe.

Erstellen Sie eine dynamische Rechenknotengruppe mit der `purchaseOption` Einstellung auf `INTERRUPTIBLE_CAPACITY_RESERVATION` und der Startvorlage, die I-ODCR auf verweist. Setzen Sie die Mindestanzahl der Instanzen auf 0 und die maximale Anzahl auf 16 (entsprechend der I-ODCR Kapazität).

Überlegungen zur Abrechnung

- Der ODCR-Besitzer zahlt On-Demand Gebühren für ungenutzte Kapazität in den I-ODCR (Instances, die nicht vom Kunden gestartet wurden).

- Der Verbraucher zahlt On-Demand Tarife nur für Instances, die er tatsächlich startet und nutzt.

Weitere Informationen finden Sie unter Kapazitätsreservierung, Preise und Abrechnung im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Verwenden von Amazon EC2 EC2-Kapazitätsblöcken für ML mit AWS PCS

Amazon EC2 Capacity Blocks for ML ist eine Amazon EC2 EC2-Kaufoption, mit der Sie im Voraus bezahlen können, um GPU-basierte Accelerated Computing-Instances innerhalb eines bestimmten Datums und Zeitbereichs zu reservieren, um Workloads mit kurzer Dauer zu unterstützen. Instances, die innerhalb eines Kapazitätsblocks ausgeführt werden, werden in Amazon EC2 automatisch nahe beieinander platziert UltraClusters, um blockierungsfreie Netzwerke im Petabit-Bereich mit niedriger Latenz zu gewährleisten. Weitere Informationen finden Sie unter Capacity Blocks for ML im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Sie können eine Startvorlage verwenden, damit AWS PCS beim Starten von Instances für eine Rechenknotengruppe einen Kapazitätsblock verwendet.

> **Note**
>
> AWS PCS hat seit Slurm-Version 24.05 Unterstützung für Capacity Blocks eingeführt.

Einschränkungen

- AWS PCS unterstützt nur Capacity-Blöcke mit den Instance-Familien P6-B300, P6-B200, P5en, P5e, P5 und P4d.
- Sie können eine Rechenknotengruppe jeweils nur einem Kapazitätsblock zuordnen.
- Sie können eine Rechenknotengruppe keiner Kapazitätsreservierungsgruppe zuordnen, die mehrere Kapazitätsblöcke kombiniert.
- Kapazitätsblöcke müssen sich im `active` Status `scheduled` oder befinden, um sie mit AWS PCS verwenden zu können. Sie können Kapazitätsblöcke nicht in anderen Zuständen verwenden, `payment-failed` z. Weitere Informationen finden Sie unter Kapazitätsblöcke anzeigen im Amazon Elastic Compute Cloud-Benutzerhandbuch.
- Informationen zu den Instance-Typen P6 und P5 finden Sie in der entsprechenden AWS-Dokumentation: Softwareanforderungen für P6-Instances, Maximieren der Netzwerkbandbreite auf Amazon EC2 EC2-Instances mit mehreren Netzwerkkarten

Ablauf des Kapazitätsblocks

Kapazitätsblöcke sind auf ein bestimmtes Datum und einen bestimmten Zeitraum beschränkt. Wenn ein Kapazitätsblock abläuft:

- Die mit diesem Kapazitätsblock verknüpfte Rechenknotengruppe ist weiterhin vorhanden und bleibt denselben Warteschlangen zugeordnet.
- Alle Instanzen in der Compute-Knotengruppe sind beendet und aktive Jobs können je nach Ihren Slurm-Einstellungen fehlschlagen.
- AWS PCS kann keine neuen Instanzen in der Compute-Knotengruppe starten.
- Alle in der Warteschlange befindlichen oder neu eingereichten Jobs verbleiben im Status „Ausstehend“, bis eine weitere Rechenknotengruppe an die Warteschlange angehängt wird oder Sie die Compute-Knotengruppe so aktualisieren, dass sie eine neue Startvorlage verwendet, die einen neuen Kapazitätsblock angibt.

Konfigurieren Sie eine AWS PCS-Rechenknotengruppe für die Verwendung eines Kapazitätsblocks

Um einen Kapazitätsblock einer Rechenknotengruppe zuzuordnen

1. Erstellen Sie eine Amazon EC2 EC2-Startvorlage für AWS PCS, die Ihren Kapazitätsblock spezifiziert. Weitere Informationen zum Erstellen einer Startvorlage für AWS PCS finden Sie unterVerwenden von Amazon EC2 EC2-Startvorlagen mit AWS PCS.

   Ihre Startvorlage muss Folgendes enthalten:

   - Der Wert `MarketType` von `InstanceMarketOptions` muss auf gesetzt sein`capacity-block`.
   - A `CapacityReservationSpecification` mit einem gültigen `CapacityReservationId`
   - Ein gültiger `InstanceType` Wert, der dem Instance-Typ des von Ihnen gekauften Kapazitätsblocks entspricht.
2. Erstellen Sie eine Compute-Knotengruppe, die die Startvorlage verwendet. Weitere Informationen finden Sie unter Erstellen einer Compute-Knotengruppe in AWS STK.. Sie können auch eine bestehende Compute-Knotengruppe aktualisieren, um die Startvorlage zu verwenden. Weitere Informationen finden Sie unter Aktualisierung eines AWS PCS-Compute-Knotengruppe.

   Wenn Sie die Compute-Knotengruppe erstellen oder aktualisieren:

- Die IAM-Identität, die Sie zum Erstellen oder Aktualisieren der Compute-Knotengruppe verwenden, muss über die folgenden Berechtigungen verfügen:

  ```
  ec2:DescribeCapacityReservations
  ec2:DescribeCapacityBlocks
  ec2:DescribeCapacityBlockStatus
  ```

  Weitere Informationen finden Sie unter Mindestberechtigungen für AWS STK..
- Der Kapazitätsblock muss sich im `active` Status `scheduled` oder befinden.
- Stellen Sie den `purchaseOption` Wert der Compute-Knotengruppe auf ein`CAPACITY_BLOCK`.
- Der Wert `maxInstanceCount` der Rechenknotengruppe darf die Größe des Kapazitätsblocks nicht überschreiten.
- Die Verfügbarkeitszone der Compute-Knotengruppe muss mit einer der Subnetz-Verfügbarkeitszonen der Compute-Knotengruppe übereinstimmen.

> **Important**
>
> Sie können den Instanztyp einer Compute-Knotengruppe nicht ändern, wenn Sie sie aktualisieren. Sie können einen Kapazitätsblock nur mit demselben Instanztyp wie die Compute-Knotengruppe verwenden. Wenn Sie einen Kapazitätsblock mit einem anderen Instanztyp verwenden möchten, müssen Sie eine neue Rechenknotengruppe erstellen.

Häufig gestellte Fragen zur Verwendung von Capacity Blocks mit AWS PCS

Ich habe gerade für einen Kapazitätsblock bezahlt und sofort versucht, ihn mit AWS PCS zu verwenden, aber die Erstellung der Compute-Knotengruppe ist fehlgeschlagen. Was ist passiert?

Ihr Kapazitätsblock befindet sich möglicherweise nicht im `active` Status `scheduled` Oder. Versuchen Sie es erneut, wenn der Kapazitätsblock den Wert `scheduled` oder hat`active`.

Ich verwende einen Capacity Block in AWS PCS und habe eine Erweiterung gekauft, bevor sie abgelaufen ist. Wie verwende ich ihn weiterhin in AWS PCS?

Sie müssen nichts tun, um den Capacity Block in AWS PCS weiterhin zu verwenden. Das Enddatum Ihres Capacity Blocks wird aktualisiert, sobald Ihre Verlängerungszahlung erfolgreich

war. Solange Ihr Kapazitätsblock nicht abläuft, ist die Rechenknotengruppe weiterhin in Betrieb. Wenn Ihre Verlängerungszahlung fehlschlägt, bleibt Ihr Kapazitätsblock bestehen `active` und die Rechenknotengruppe funktioniert, bis der Kapazitätsblock an seinem ursprünglichen Enddatum abläuft.

Was passiert mit meinen in der Warteschlange stehenden und laufenden Jobs, wenn mein Kapazitätsblock abläuft?

Jobs in der Warteschlange, die nicht gestartet wurden, bevor der Kapazitätsblock abgelaufen ist, bleiben solange ausstehend, bis Sie eine weitere Rechenknotengruppe an die Warteschlange anhängen oder die Rechenknotengruppe mit einem neuen Kapazitätsblock aktualisieren. Sie können weiterhin Jobs an die Warteschlange senden. Ihre Slurm-Einstellungen wirken sich auf aktive Jobs aus. Standardmäßig werden aktive Jobs automatisch erneut in die Warteschlange gestellt, können aber Fehler aufweisen oder fehlschlagen.

Mein Kapazitätsblock ist abgelaufen. Sollte ich etwas tun?

Du musst nichts tun. Sie können in der Amazon EC2 EC2-Konsole den Status Ihrer EC2-Kapazitätsreservierungen überprüfen. Wenn ein Kapazitätsblock abläuft, ist die diesem Kapazitätsblock zugeordnete Rechenknotengruppe weiterhin vorhanden und verarbeitet dieselben Warteschlangen. Die Rechenknotengruppe hat keine Instanzen zum Ausführen von Jobs. Sie können die Compute-Knotengruppe löschen oder sie von den Warteschlangen trennen, um zu verhindern, dass Benutzer Jobs einreichen, die nicht ausgeführt werden können.

Ich möchte einen neuen Kapazitätsblock mit meiner AWS PCS-Compute-Knotengruppe verwenden. Was soll ich tun?

Wir empfehlen Ihnen, eine neue Rechenknotengruppe zu erstellen, um den neuen Kapazitätsblock zu verwenden. Weitere Informationen finden Sie unter Konfigurieren Sie eine AWS PCS-Rechenknotengruppe für die Verwendung eines Kapazitätsblocks.

Wie kann ich einen Kapazitätsblock für Cluster und Dienste gemeinsam nutzen?

Sie können einen Kapazitätsblock auf mehrere Cluster und Dienste aufteilen. Um beispielsweise einen Kapazitätsblock mit 64 `p5.48xlarge` Instanzen mit 20 Knoten auf PCS-Cluster-1, 16 Knoten auf PCS-Cluster-2 und den verbleibenden Knoten für andere Dienste aufzuteilen, setzen Sie beide und auf 20 für PCS-Cluster-1 `minInstanceCount` und `maxInstanceCount` 16 für PCS-Cluster-2.

Kann ich mehr als einen Kapazitätsblock oder kombinierte Kapazität mit einer Rechenknotengruppe verwenden?

Nein. Einer einzelnen Rechenknotengruppe kann nur 1 Kapazitätsblock zugeordnet werden. AWS PCS unterstützt keine Kapazitätsreservierungsgruppen, die mehrere Kapazitätsblöcke kombinieren.

Woher weiß ich, wann meine Kapazitätsblöcke beginnen oder ablaufen?

Unabhängig von AWS PCS sendet Amazon EC2 ein `Capacity Block Reservation Delivered` Ereignis, EventBridge wenn eine Kapazitätsblock-Reservierung beginnt, und ein `Capacity Block Reservation Expiration Warning` Ereignis 40 Minuten vor Ablauf der Kapazitätsblock-Reservierung. Weitere Informationen finden Sie unter Verwendung von Kapazitätsblöcken überwachen EventBridge im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Wie verfolgt Slurm den Status meines Kapazitätsblocks?

Du kannst laufen`sinfo`, um zu verstehen, wie AWS PCS den Capacity Block verwendet. In der folgenden Beispielausgabe ist eine Warteschlange einer Rechenknotengruppe zugeordnet, die 4 Instanzen aus einem `active` Kapazitätsblock ausführt. Die Knoten befinden sich im `idle` Slurm-Status (zur Verwendung verfügbar und noch keinen Jobs zugewiesen).

```
$ sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
fanout up infinite 4 idle node-fanout-[1-4]
```

Wenn sich die Knoten stattdessen im `maint` Status befinden, können Sie den Befehl ausführen, `scontrol show res` um Details zur Slurm-Reservierung zu sehen, die diesen Status kontrolliert. In der folgenden Beispielausgabe hat der `scheduled` Capacity-Block ein future Startdatum.

```
$ scontrol show res

ReservationName=node-fanout-scheduled StartTime=2025-10-14T13:09:17
 EndTime=2025-10-14T13:11:17 Duration=00:02:00
   Nodes=node-fanout-[1-4] NodeCnt=4 CoreCnt=16 Features=(null) PartitionName=(null)
 Flags=MAINT,SPEC_NODES
   TRES=cpu=16

   Users=root Groups=(null) Accounts=(null) Licenses=(null) State=ACTIVE
 BurstBuffer=(null)
```

```
   MaxStartDelay=(null)

   Comment=node-fanout Scheduled
```

Wie kann ich feststellen, ob die Fehler, die ich beim Starten von Capacity erhalte, darauf zurückzuführen sind, dass mein Capacity-Block gemeinsam genutzt wird?

Überprüfen Sie Capacity Reservations in der Amazon EC2 EC2-Konsole, um herauszufinden, wie viele Instances aus dem Capacity Block aktiv bereitgestellt werden. Überprüfen Sie die Tags der einzelnen Instances, um herauszufinden, welcher Service oder Cluster sie verwendet. Beispielsweise verfügen alle Instanzen für AWS PCS über AWS PCS-Tags, `aws:pcs:cluster-id = pcs_l0mizqyk5o | aws:pcs:compute-node-group-id = pcs_ic7onkmfqk` die angeben, zu welchen Clustern und Rechenknotengruppen die Instanz gehört. Sie können dann überprüfen, ob der Kapazitätsblock die maximale Kapazität erreicht hat.

Sie verwenden`scontrol show nodes`, um zu überprüfen, ob ein Capacity Block-Knoten in einem AWS PCS-Cluster Folgendes auslöst`ReservationCapacityExceeded`:

```
[root@ip-172-16-10-54 ~]# scontrol show nodes test-node-8-gamma-cb-2
NodeName=test-8-gamma-cb-2 CoresPerSocket=1
   CPUAlloc=0 CPUEfctv=8 CPUTot=8 CPULoad=0.00
   AvailableFeatures=test-8-gamma-cb,gpu
   ActiveFeatures=test-8-gamma-cb,gpu
   Gres=gpu:H100:1
   NodeAddr=test-8-gamma-cb-2 NodeHostName=test-8-gamma-cb-2
   RealMemory=249036 AllocMem=0 FreeMem=N/A Sockets=8 Boards=1
   State=IDLE+CLOUD+POWERING_DOWN ThreadsPerCore=1 TmpDisk=0 Weight=1 Owner=N/A
 MCS_label=N/A
   Partitions=my-q
   BootTime=None SlurmdStartTime=None
   LastBusyTime=Unknown ResumeAfterTime=None
   CfgTRES=cpu=8,mem=249036M,billing=8
   AllocTRES=
   CurrentWatts=0 AveWatts=0
   Reason=Failed to launch backing instance (Error Code:
 ReservationCapacityExceeded) [root@2025-08-28T15:15:33]
```

Wie kann ich erzwingen, dass ein Job auf Capacity Block-gestützten Instances ausgeführt wird, wenn mehrere Rechenknotengruppen an dieselbe Warteschlange angehängt sind?

Sie können die Funktionen und Einschränkungen von Slurm verwenden, um einen Job an eine bestimmte Gruppe von Knoten zu binden. Wir empfehlen, Slurm-Gewichtungen nicht für jede

Rechenknotengruppe festzulegen, da dies nur mit Knoten funktioniert, die sich nicht im `maint` Status befinden.

Nützliche Parameter für Startvorlagen

In diesem Abschnitt werden einige Parameter für Startvorlagen beschrieben, die für AWS PCS allgemein nützlich sein können.

Aktivieren Sie die detaillierte CloudWatch Überwachung

Mithilfe eines Startvorlagenparameters können Sie die Erfassung von CloudWatch Metriken in kürzeren Intervallen aktivieren.

AWS-Managementkonsole

Auf den Konsolenseiten zum Erstellen oder Bearbeiten von Startvorlagen befindet sich diese Option im Abschnitt Erweiterte Details. Stellen Sie „Detaillierte CloudWatch Überwachung“ auf „Aktivieren“.

YAML

```
Monitoring:
    Enabled: True
```

JSON

```
{"Monitoring": {"Enabled": "True"}}
```

Weitere Informationen finden Sie unter Aktivieren oder Deaktivieren der detaillierten Überwachung für Ihre Instances im Amazon Elastic Compute Cloud-Benutzerhandbuch für Linux-Instances.

Instanz-Metadaten-Service Version 2 (IMDS v2)

Die Verwendung von IMDS v2 mit EC2-Instances bietet erhebliche Sicherheitsverbesserungen und trägt dazu bei, potenzielle Risiken im Zusammenhang mit dem Zugriff auf Instance-Metadaten in Umgebungen zu minimieren. AWS

AWS-Managementkonsole

Auf den Konsolenseiten zum Erstellen oder Bearbeiten von Startvorlagen befindet sich diese Option im Abschnitt Erweiterte Details. Stellen Sie für Metadaten, auf die zugegriffen werden kann, die Option Aktiviert, die Metadatenversion auf Nur V2 (Token erforderlich) und das Limit für den Metadaten-Response-Hop auf 4 ein.

YAML

```
MetadataOptions:
  HttpEndpoint: enabled
  HttpTokens: required
  HttpPutResponseHopLimit: 4
```

JSON

```
{
    "MetadataOptions": {
        "HttpEndpoint": "enabled",
        "HttpPutResponseHopLimit": 4,
        "HttpTokens": "required"
    }
}
```

.

AWS PCS-Warteschlangen

Eine AWS PCS-Warteschlange ist eine einfache Abstraktion gegenüber der systemeigenen Implementierung einer Arbeitswarteschlange durch den Scheduler. Im Fall von Slurm entspricht eine AWS PCS-Warteschlange einer Slurm-Partition.

Benutzer senden Jobs an eine Warteschlange, in der sie sich befinden, bis sie so geplant werden können, dass sie auf Knoten ausgeführt werden, die von einer oder mehreren Rechenknotengruppen bereitgestellt werden. Ein AWS PCS-Cluster kann mehrere Jobwarteschlangen haben. Sie können beispielsweise eine Warteschlange erstellen, die Amazon EC2 On-Demand-Instances für Jobs mit hoher Priorität verwendet, und eine weitere Warteschlange, die Amazon EC2 Spot-Instances für Jobs mit niedriger Priorität verwendet.

Themen

Eine Warteschlange in AWS PCS erstellen

Dieses Thema bietet einen Überblick über die verfügbaren Optionen und beschreibt, was beim Erstellen einer Warteschlange in AWS PCS zu beachten ist.

> **Note**
>
> Sie können benutzerdefinierte Slurm-Einstellungen für Warteschlangen konfigurieren, um partitionsspezifische Planungsrichtlinien und Ressourcenverwaltung zu implementieren. Weitere Informationen finden Sie unter Konfiguration benutzerdefinierter Slurm-Einstellungen in AWS STK..

Voraussetzungen

- Ein AWS PCS-Cluster — Warteschlangen können nur in Verbindung mit einem bestimmten PCS-Cluster erstellt werden. AWS

- Eine oder mehrere AWS PCS-Compute-Knotengruppen — eine Warteschlange muss mindestens einer AWS PCS-Compute-Knotengruppe zugeordnet sein.

Um eine Warteschlange in AWS PCS zu erstellen

Sie können eine Warteschlange mit dem AWS-Managementkonsole oder dem erstellen AWS CLI.

AWS-Managementkonsole

Um eine Warteschlange mit der Konsole zu erstellen

1. Öffnen Sie die AWS PCS-Konsole.
2. Wählen Sie den Cluster für die Warteschlange aus. Navigieren Sie zu Warteschlangen und wählen Sie Warteschlange erstellen.
3. Geben Sie im Abschnitt Warteschlangenkonfiguration die folgenden Werte an:
    a. Warteschlangenname — Ein Name für Ihre Warteschlange. Der Name darf nur alphanumerische Zeichen (wobei die Groß- und Kleinschreibung beachtet werden muss) und Bindestriche enthalten. Er muss mit einem alphabetischen Zeichen beginnen und darf nicht länger als 25 Zeichen sein. Der Name muss innerhalb des Clusters eindeutig sein.
    b. Compute-Knotengruppen — Wählen Sie eine oder mehrere Compute-Knotengruppen aus, um diese Warteschlange zu bedienen. Eine Rechenknotengruppe kann mehr als einer Warteschlange zugeordnet werden.
4. (Optional) Im Abschnitt Zusätzliche Scheduler-Einstellungen können Sie Parameternamen- und Wertepaare hinzufügen, um zusätzliche Slurm-Einstellungen zu konfigurieren. Eine vollständige Liste der unterstützten Parameter finden Sie unter. Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Warteschlangen
5. (Optional) Fügen Sie unter Tags beliebige Tags zu Ihrer AWS PCS-Warteschlange hinzu
6. Wählen Sie Create queue (Warteschlange erstellen) aus. Im Statusfeld wird Creating angezeigt, während AWS PCS die Warteschlange erstellt. Die Erstellung der Warteschlange kann mehrere Minuten dauern.

Als nächster Schritt wird empfohlen

- Reichen Sie einen Job in Ihre neue Warteschlange ein.

AWS CLI

Um eine Warteschlange zu erstellen mit AWS CLI

Verwenden Sie den folgenden Befehl, um Ihre Warteschlange zu erstellen. Nehmen Sie die folgenden Ersetzungen vor:

1. *region-code*Ersetzen Sie durch die AWS Region des Clusters. Beispiel, `us-east-1`.
2. *my-queue*Ersetzen Sie es durch den Namen für Ihre Warteschlange. Der Name darf nur alphanumerische Zeichen (wobei die Groß- und Kleinschreibung beachtet werden muss) und Bindestriche enthalten. Er muss mit einem alphabetischen Zeichen beginnen und darf nicht länger als 25 Zeichen sein. Der Name muss innerhalb des Clusters eindeutig sein.
3. *my-cluster*Ersetzen Sie ihn durch den Namen oder die ID Ihres Clusters.
4. *compute-node-group-id*Ersetzen Sie es durch die ID der Rechenknotengruppe, die die Warteschlange bedienen soll. Beispiel, `pcs_abcdef12345`.

> **Note**
>
> Wenn Sie eine Warteschlange erstellen, müssen Sie die ID der Compute-Knotengruppe und nicht deren Namen angeben.

```
aws pcs create-queue --region region-code \
    --queue-name my-queue \
    --cluster-identifier my-cluster \
    --compute-node-group-configurations \
    computeNodeGroupId=compute-node-group-id
```

Example— Erstellen einer Warteschlange mit benutzerdefinierten Slurm-Einstellungen

```
aws pcs create-queue --region region-code \
    --queue-name my-queue \
    --cluster-identifier my-cluster \
    --compute-node-group-configurations \
    computeNodeGroupId=compute-node-group-id \
    --slurm-configuration \
    'slurmCustomSettings=[{parameterName=Default,parameterValue=YES}]'
```

Weitere Informationen finden Sie unter Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Warteschlangen.

Das Erstellen der Warteschlange kann mehrere Minuten dauern. Sie können den Status Ihrer Warteschlange mit dem folgenden Befehl abfragen. Sie können keine Jobs an die Warteschlange senden, bis ihr Status erreicht ist`ACTIVE`.

```
aws pcs get-queue --region region-code \
    --cluster-identifier my-cluster \
    --queue-identifier my-queue
```

Als nächster Schritt wird empfohlen

- Reichen Sie einen Job in Ihre neue Warteschlange ein

Aktualisierung einer AWS PCS-Warteschlange

Dieses Thema bietet einen Überblick über die verfügbaren Optionen und beschreibt, was bei der Aktualisierung einer AWS PCS-Warteschlange zu beachten ist. Informationen zu den benutzerdefinierten Slurm-Einstellungen finden Sie unterBenutzerdefinierte Slurm-Einstellungen für AWS PCS-Warteschlangen.

Überlegungen beim Aktualisieren einer AWS PCS-Warteschlange

Warteschlangenaktualisierungen wirken sich nicht auf laufende Jobs aus, aber der Cluster kann möglicherweise keine neuen Jobs annehmen, während die Warteschlange aktualisiert wird.

Um eine AWS PCS-Warteschlange zu aktualisieren

Sie können das AWS-Managementkonsole oder verwenden AWS CLI , um eine Warteschlange zu aktualisieren.

AWS-Managementkonsole

Um eine Warteschlange zu aktualisieren

1. Öffnen Sie die AWS PCS-Konsole unter `https://console.aws.amazon.com/pcs/home#/clusters`
2. Wählen Sie den Cluster aus, in dem Sie eine Warteschlange aktualisieren möchten.

3. Navigieren Sie zu Warteschlangen, gehen Sie zu der Warteschlange, die Sie aktualisieren möchten, und wählen Sie dann Bearbeiten aus.
4. Aktualisieren Sie im Abschnitt Warteschlangenkonfiguration einen der folgenden Werte:
   - Knotengruppen — Fügen Sie Compute-Knotengruppen hinzu oder entfernen Sie sie aus der Zuordnung zur Warteschlange.
   - Zusätzliche Scheduler-Einstellungen — Fügen Sie benutzerdefinierte Slurm-Einstellungen für die Warteschlange hinzu, ändern oder entfernen Sie sie. Weitere Informationen finden Sie unter Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Warteschlangen.
   - Tags — Fügen Sie Tags für die Warteschlange hinzu oder entfernen Sie sie.
5. Wählen Sie Aktualisieren aus. Im Feld Status wird die Meldung Aktualisierung angezeigt, während die Änderungen übernommen werden.

> **Important**
>
> Aktualisierungen in der Warteschlange können mehrere Minuten dauern.

AWS CLI

Um eine Warteschlange zu aktualisieren

1. Aktualisieren Sie Ihre Warteschlange mit dem folgenden Befehl. Nehmen Sie vor der Ausführung des Befehls die folgenden Ersetzungen vor:
   a. *region-code* Ersetzen Sie es durch AWS-Region das, in dem Sie Ihren Cluster erstellen möchten.
   b. *my-queue* Ersetzen Sie es durch den Namen oder `computeNodeGroupId` für Ihre Warteschlange.
   c. *my-cluster* Ersetzen Sie durch den Namen oder `clusterId` Ihres Clusters.
   d. Um die Zuordnungen von Compute-Knotengruppen zu ändern, stellen Sie eine aktualisierte Liste für bereit `--compute-node-group-configurations`.
      - Um beispielsweise eine zweite Compute-Knotengruppe hinzuzufügen `computeNodeGroupExampleID2`:

```
--compute-node-group-configurations
 computeNodeGroupId=computeNodeGroupExampleID1,computeNodeGroupId=computeNodeGro
```

```
aws pcs update-queue --region region-code \
    --queue-identifier my-queue \
    --cluster-identifier my-cluster \
    --compute-node-group-configurations \
    computeNodeGroupId=computeNodeGroupExampleID1
```

Example— Aktualisierung einer Warteschlange mit benutzerdefinierten Slurm-Einstellungen

```
aws pcs update-queue --region region-code \
    --queue-identifier my-queue \
    --cluster-identifier my-cluster \
    --slurm-configuration \
    'slurmCustomSettings=[{parameterName=Default,parameterValue=YES}]'
```

Weitere Informationen finden Sie unter Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Warteschlangen.

2. Das Aktualisieren der Warteschlange kann mehrere Minuten dauern. Sie können den Status Ihrer Warteschlange mit dem folgenden Befehl abfragen. Sie können keine Jobs an die Warteschlange senden, bis ihr Status erreicht ist`ACTIVE`.

```
aws pcs get-queue --region region-code \
    --cluster-identifier my-cluster \
    --queue-identifier my-queue
```

Empfohlene nächste Schritte

- Reichen Sie einen Job in Ihre aktualisierte Warteschlange ein.

Löschen einer Warteschlange in AWS PCS

Dieses Thema bietet einen Überblick über das Löschen einer Warteschlange in AWS PCS.

Überlegungen beim Löschen einer Warteschlange

- Wenn in der Warteschlange Jobs ausgeführt werden, werden sie vom Scheduler beendet, wenn die Warteschlange gelöscht wird. Ausstehende Jobs in der Warteschlange werden storniert. Erwägen Sie, darauf zu warten, dass Jobs in der Warteschlange abgeschlossen sind, oder stop/cancel sie manuell mit den systemeigenen Befehlen des Schedulers (z. B. `scancel` für Slurm) zu bearbeiten.

Lösche die Warteschlange

Sie können das AWS-Managementkonsole oder verwenden AWS CLI , um eine Warteschlange zu löschen.

AWS-Managementkonsole

So löschen Sie eine Warteschlange

1. Öffnen Sie die AWS PCS-Konsole.
2. Wählen Sie den Cluster der Warteschlange aus.
3. Navigieren Sie zu Warteschlangen und wählen Sie die Warteschlange aus, die Sie löschen möchten.
4. Wählen Sie Löschen aus.
5. Das Feld Status wird angezeigt`Deleting`. Das kann mehrere Minuten dauern.

> **Note**
>
> Sie können Befehle verwenden, die Ihrem Scheduler eigen sind, um zu bestätigen, dass die Warteschlange gelöscht wurde. Verwenden Sie zum Beispiel `sinfo` or `squeue` für Slurm.

AWS CLI

So löschen Sie eine Warteschlange

- Verwenden Sie den folgenden Befehl, um eine Warteschlange mit diesen Ersetzungen zu löschen:

- Ersetzen Sie *region-code* durch den, in dem sich AWS-Region Ihr Cluster befindet.
- *my-queue*Ersetzen Sie durch den Namen oder die ID Ihrer Warteschlange.
- *my-cluster*Ersetzen Sie durch den Namen oder die ID Ihres Clusters.

```
aws pcs delete-queue --region region-code \
       --queue-identifier my-queue \
       --cluster-identifier my-cluster
```

Das Löschen der Warteschlange kann mehrere Minuten dauern.

> **Note**
>
> Sie können Befehle verwenden, die Ihrem Scheduler eigen sind, um zu bestätigen, dass die Warteschlange gelöscht wurde. Verwenden Sie zum Beispiel `sinfo` or `squeue` für Slurm.

AWS PCS-Anmeldeknoten

Ein AWS PCS-Cluster benötigt normalerweise mindestens einen Anmeldeknoten, um den interaktiven Zugriff und die Auftragsverwaltung zu unterstützen. Eine Möglichkeit, dies zu erreichen, besteht darin, eine statische AWS PCS-Rechenknotengruppe zu verwenden, die für die Funktion eines Anmeldeknotens konfiguriert ist. Sie können auch eine eigenständige EC2-Instance so konfigurieren, dass sie als Anmeldeknoten fungiert.

Themen

Verwendung einer AWS PCS-Compute-Knotengruppe zur Bereitstellung von Anmeldeknoten

Dieses Thema bietet einen Überblick über vorgeschlagene Konfigurationsoptionen und beschreibt, was zu beachten ist, wenn Sie eine AWS PCS-Rechenknotengruppe verwenden, um dauerhaften, interaktiven Zugriff auf Ihren Cluster bereitzustellen.

Erstellen einer AWS PCS-Rechenknotengruppe für Anmeldeknoten

Operativ unterscheidet sich dies nicht wesentlich von der Erstellung einer regulären Rechenknotengruppe. Es müssen jedoch einige wichtige Konfigurationsentscheidungen getroffen werden:

- Legen Sie eine statische Skalierungskonfiguration für mindestens eine EC2-Instance in der Compute-Knotengruppe fest.
- Wählen Sie die On-Demand-Kaufoption, um zu vermeiden, dass Ihre Instance (s) zurückgefordert werden.
- Wählen Sie einen aussagekräftigen Namen für die Compute-Knotengruppe, z. B. Login.
- Wenn Sie möchten, dass auf die Login-Knoten-Instanz (en) außerhalb Ihrer VPC zugegriffen werden kann, sollten Sie die Verwendung eines öffentlichen Subnetzes in Betracht ziehen.
- Wenn Sie den SSH-Zugriff zulassen möchten, muss die Startvorlage über eine Sicherheitsgruppe verfügen, die den SSH-Port den IP-Adressen Ihrer Wahl zugänglich macht.

- Das IAM-Instance-Profil sollte nur die AWS-Berechtigungen haben, die Ihre Endbenutzer haben sollen. Details dazu finden Sie unter IAM-Instanzprofile für AWS Dienst für parallele Datenverarbeitung.
- Erwägen Sie, AWS Systems Manager Session Manager die Verwaltung Ihrer Login-Instances zu gestatten.
- Erwägen Sie, den Zugriff auf die AWS-Anmeldeinformationen der Instanz auf Administratorbenutzer zu beschränken
- Wählen Sie kostengünstigere Instance-Typen als für reguläre Compute-Knotengruppen aus, da die Login-Knoten kontinuierlich laufen.
- Verwenden Sie dasselbe (oder ein abgeleitetes) AMI wie für Ihre anderen Compute-Knotengruppen, um sicherzustellen, dass auf allen Instances dieselbe Software installiert ist. Weitere Informationen zum Anpassen finden Sie AMIs unter Amazon Machine Images (AMIs) für AWS STK.
- Konfigurieren Sie dasselbe Netzwerkdateisystem (Amazon EFS, Amazon FSx for Lustre usw.), das auf Ihren Anmeldeknoten bereitgestellt wird wie auf Ihren Compute-Instances. Weitere Informationen finden Sie unter Verwenden von Netzwerkdateisystemen mit AWS PCS.

Greifen Sie auf Ihre Anmeldeknoten zu

Sobald Ihre neue Compute-Knotengruppe den Status ACTIVE erreicht hat, können Sie die EC2-Instance (en) finden, die sie erstellt hat, und sich bei ihnen anmelden. Weitere Informationen finden Sie unter Suchen nach Rechenknotengruppeninstanzen in AWS PCS.

Aktualisierung einer AWS PCS-Compute-Knotengruppe für Login-Knoten

Sie können eine Anmeldeknotengruppe aktualisieren mit UpdateComputeNodeGroup. Im Rahmen des Aktualisierungsprozesses für Knotengruppen werden laufende Instanzen ersetzt. Beachten Sie, dass dadurch alle aktiven Benutzersitzungen oder Prozesse auf der Instanz unterbrochen werden. Laufende oder in der Warteschlange befindliche Slurm-Jobs sind davon nicht betroffen. Weitere Informationen finden Sie unter Aktualisierung eines AWS PCS-Compute-Knotengruppe.

Sie können auch die Startvorlage bearbeiten, die von Ihrer Compute-Knotengruppe verwendet wird. Sie müssen sie verwenden UpdateComputeNodeGroup , um die aktualisierte Startvorlage auf die Compute-Knotengruppe anzuwenden. Neue EC2-Instances, die in der Compute-Knotengruppe gestartet werden, verwenden die aktualisierte Startvorlage. Weitere Informationen finden Sie unter Verwenden von Amazon EC2 EC2-Startvorlagen mit AWS PCS.

Löschen einer AWS PCS-Compute-Knotengruppe für Anmeldeknoten

Sie können eine Anmeldeknotengruppe mithilfe des Mechanismus zum Löschen von Compute-Knotengruppen in AWS PCS aktualisieren. Laufende Instanzen werden im Rahmen des Löschens der Knotengruppe beendet. Bitte beachten Sie, dass dadurch alle aktiven Benutzersitzungen oder Prozesse auf der Instanz unterbrochen werden. Laufende oder in der Warteschlange befindliche Slurm-Jobs sind davon nicht betroffen. Weitere Informationen finden Sie unter Löschen einer Compute-Knotengruppe in AWS PCS.

Verwenden eigenständiger Instanzen als AWS PCS-Anmeldeknoten

Sie können unabhängige EC2-Instances einrichten, um mit dem Slurm-Scheduler eines AWS PCS-Clusters zu interagieren. Dies ist nützlich, um Anmeldeknoten, Workstations oder dedizierte Workflow-Management-Hosts zu erstellen, die mit AWS PCS-Clustern funktionieren, aber außerhalb des PCS-Managements betrieben werden. AWS Dazu muss jede eigenständige Instanz:

1. Eine kompatible Slurm-Softwareversion installiert haben.
2. In der Lage sein, eine Verbindung zum AWS Slurmctld-Endpunkt des PCS-Clusters herzustellen.
3. Sorgen Sie dafür, dass Slurm Auth und Cred Kiosk Daemon (`sackd`) ordnungsgemäß mit dem Endpunkt und dem Secret des PCS-Clusters konfiguriert sind. AWS Weitere Informationen finden Sie unter sackd in der Slurm-Dokumentation.

Dieses Tutorial hilft Ihnen bei der Konfiguration einer unabhängigen Instanz, die eine Verbindung zu einem AWS PCS-Cluster herstellt.

Inhalt

Schritt 1 — Rufen Sie die Adresse und das Geheimnis für den AWS Ziel-PCS-Cluster ab

Rufen Sie mithilfe des folgenden Befehls Details zum AWS AWS CLI Ziel-PCS-Cluster ab. Nehmen Sie vor der Ausführung des Befehls die folgenden Ersetzungen vor:

- *region-code*Ersetzen Sie durch den AWS-Region Ort, an dem der Zielcluster ausgeführt wird.
- *cluster-ident*Ersetzen Sie durch den Namen oder die ID für den Zielcluster

```
aws pcs get-cluster --region region-code --cluster-identifier cluster-ident
```

Der Befehl gibt eine Ausgabe zurück, die diesem Beispiel ähnelt.

```
{
    "cluster": {
        "name": "get-started",
        "id": "pcs_123456abcd",
        "arn": "arn:aws:pcs:us-east-1:111122223333:cluster/pcs_123456abcd",
        "status": "ACTIVE",
        "createdAt": "2024-12-17T21:03:52+00:00",
        "modifiedAt": "2024-12-17T21:03:52+00:00",
        "scheduler": {
            "type": "SLURM",
            "version": "25.11"
        },
        "size": "SMALL",
        "slurmConfiguration": {
            "authKey": {
                "secretArn": "arn:aws:secretsmanager:us-east-1:111122223333:secret:pcs!
slurm-secret-pcs_123456abcd-a12ABC",
                "secretVersion": "ef232370-d3e7-434c-9a87-ec35c1987f75"
            }
        },
        "networking": {
            "subnetIds": [
                "subnet-0123456789abcdef0"
            ],
            "securityGroupIds": [
                "sg-0123456789abcdef0"
            ]
```

```
        },
        "endpoints": [
            {
                "type": "SLURMCTLD",
                "privateIpAddress": "10.3.149.220",
                "port": "6817"
            }
        ]
    }
}
```

In diesem Beispiel hat der Cluster-Slurm-Controller-Endpunkt die IP-Adresse `10.3.149.220` und er läuft auf dem Port`6817`. Der `secretArn` wird in späteren Schritten verwendet, um das Clustergeheimnis abzurufen. Die IP-Adresse und der Port werden in späteren Schritten zur Konfiguration des `sackd` Dienstes verwendet.

Schritt 2 — Starten Sie eine EC2-Instanz

Starten Sie EC2-Instances wie folgt:

1. Öffnen Sie die Amazon EC2-Konsole.
2. Wählen Sie im Navigationsbereich Instances und dann Instances starten aus, um den Launch Instance Wizard zu öffnen.
3. (Optional) Geben Sie im Abschnitt Name und Tags einen Namen für die Instance ein, z. B. `PCS-LoginNode` Der Name wird der Instance als Ressourcen-Tag (`Name=PCS-LoginNode`) zugewiesen.
4. Wählen Sie im Abschnitt Anwendungs- und Betriebssystemimages ein AMI für eines der von AWS PCS unterstützten Betriebssysteme aus. Weitere Informationen finden Sie unter Unterstützte Betriebssysteme.
5. Wählen Sie im Bereich Instance-Typ einen unterstützten Instance-Typ aus. Weitere Informationen finden Sie unter Unterstützte Instance-Typen.
6. Wählen Sie im Abschnitt key pair das SSH-Schlüsselpaar aus, das für die Instance verwendet werden soll.
7. Gehen Sie im Abschnitt Netzwerkeinstellungen wie folgt vor:
    - Wählen Sie Bearbeiten aus.
        i. Wählen Sie die VPC Ihres AWS PCS-Clusters aus.

ii. Für Firewall (Sicherheitsgruppen) wählen Sie Eine vorhandene Sicherheitsgruppe auswählen aus.

A. Wählen Sie eine Sicherheitsgruppe aus, die den Datenverkehr zwischen der Instanz und dem Slurm-Controller des AWS Ziel-PCS-Clusters zulässt. Weitere Informationen finden Sie unter Anforderungen und Überlegungen zur Sicherheitsgruppe.

B. (Optional) Wählen Sie eine Sicherheitsgruppe aus, die eingehenden SSH-Zugriff auf Ihre Instance ermöglicht.

8. Konfigurieren Sie im Bereich Speicher die Speichervolumes nach Bedarf. Stellen Sie sicher, dass ausreichend Speicherplatz für die Installation von Anwendungen und Bibliotheken konfiguriert ist, um Ihren Anwendungsfall zu unterstützen.
9. Wählen Sie unter Erweitert eine IAM-Rolle aus, die den Zugriff auf das Clustergeheimnis ermöglicht. Weitere Informationen finden Sie unter Holen Sie sich das Geheimnis des Slurm-Clusters.
10. Wählen Sie im Übersichtsbereich die Option Launch instance aus.

Schritt 3 — Installieren Sie Slurm auf der Instanz

Wenn die Instanz gestartet wurde und aktiv wird, stellen Sie über Ihren bevorzugten Mechanismus eine Verbindung zu ihr her. Verwenden Sie das von bereitgestellte Slurm-Installationsprogramm AWS , um Slurm auf der Instanz zu installieren. Weitere Informationen finden Sie unter Slurm-Installationsprogramm.

Laden Sie das Slurm-Installationsprogramm herunter, dekomprimieren Sie es und verwenden Sie das `installer.sh` Skript, um Slurm zu installieren. Weitere Informationen finden Sie unter Schritt 3 — Slurm installieren.

Schritt 4 — Rufen Sie das Cluster-Geheimnis ab und speichern Sie es

Diese Anweisungen erfordern die AWS CLI. Weitere Informationen finden Sie unter Installation oder Aktualisierung auf die neueste Version von AWS CLI im AWS Command Line Interface Benutzerhandbuch für Version 2.

Speichern Sie das Clustergeheimnis mit den folgenden Befehlen.

- Erstellen Sie das Konfigurationsverzeichnis für Slurm.

```
sudo mkdir -p /etc/slurm
sudo chmod 0755 /etc/slurm
```

> **Note**
>
> Durch die Einstellung von Verzeichnisberechtigungen `0755` wird sichergestellt, dass der `slurm` Benutzer das Verzeichnis durchqueren kann, um auf die Schlüsseldatei zuzugreifen. Einige Systeme verfügen möglicherweise über eine restriktive Umask, die standardmäßig Verzeichnisse mit restriktiveren Berechtigungen erstellt.

- Rufen Sie das Clustergeheimnis ab, dekodieren Sie es und speichern Sie es. Bevor Sie diesen Befehl ausführen, *`region-code`* ersetzen Sie ihn durch die Region, in der der Zielcluster ausgeführt wird, und *`secret-arn`* durch den in Schritt 1 `secretArn` abgerufenen Wert.

```
aws secretsmanager get-secret-value \
 --region region-code \
 --secret-id 'secret-arn' \
 --version-stage AWSCURRENT \
 --query 'SecretString' \
 --output text | base64 -d | sudo tee /etc/slurm/slurm.key
```

> **Warning**
>
> In einer Mehrbenutzerumgebung kann möglicherweise jeder Benutzer mit Zugriff auf die Instance das Clustergeheimnis abrufen, wenn er auf den Instance-Metadatendienst (IMDS) zugreifen kann. Dies wiederum könnte es ihnen ermöglichen, sich als andere Benutzer auszugeben. Erwägen Sie, den Zugriff auf IMDS nur auf Root- oder Administratorbenutzer zu beschränken. Erwägen Sie alternativ, einen anderen Mechanismus zu verwenden, der sich nicht auf das Instanzprofil stützt, um den geheimen Schlüssel abzurufen und zu konfigurieren.

- Legen Sie den Besitz und die Berechtigungen für die Slurm-Schlüsseldatei fest.

```
sudo chmod 0600 /etc/slurm/slurm.key
sudo chown slurm:slurm /etc/slurm/slurm.key
```

> **Note**
>
> Der Slurm-Schlüssel muss dem Benutzer und der Gruppe gehören, unter denen der `sackd` Dienst ausgeführt wird.

Schritt 5 — Konfigurieren Sie die Verbindung zum AWS PCS-Cluster

Gehen Sie wie folgt vor, um eine Verbindung zum AWS PCS-Cluster herzustellen, indem Sie ihn `sackd` als Systemdienst starten.

> **Note**
>
> Wenn Sie Slurm 25.05 oder höher verwenden, können Sie stattdessen ein Skript verwenden, um Ihren Anmeldeknoten so einzurichten, dass er eine Verbindung zu mehreren Clustern herstellt. Weitere Informationen finden Sie unter Einen eigenständigen Anmeldeknoten mit mehreren Clustern in AWS PCS verbinden.

1. Richten Sie die Umgebungsdatei für den `sackd` Dienst mit dem folgenden Befehl ein. Bevor Sie den Befehl ausführen, ersetzen Sie *`ip-address`* und *`port`* durch die in Schritt 1 von den Endpunkten abgerufenen Werte.

   ```
   sudo echo "SACKD_OPTIONS='--conf-server=ip-address:port'" > /etc/sysconfig/sackd
   ```

2. Erstellen Sie eine `systemd` Servicedatei für die Verwaltung des `sackd` Prozesses.

   ```
   sudo cat << EOF > /etc/systemd/system/sackd.service
   [Unit]
   Description=Slurm auth and cred kiosk daemon
   After=network-online.target remote-fs.target
   Wants=network-online.target
   ConditionPathExists=/etc/sysconfig/sackd

   [Service]
   Type=notify
   EnvironmentFile=/etc/sysconfig/sackd
   User=slurm
   Group=slurm
   ```

```
RuntimeDirectory=slurm
RuntimeDirectoryMode=0755
ExecStart=/opt/aws/pcs/scheduler/slurm-25.11/sbin/sackd --systemd \$SACKD_OPTIONS
ExecReload=/bin/kill -HUP \$MAINPID
KillMode=process
LimitNOFILE=131072
LimitMEMLOCK=infinity
LimitSTACK=infinity

[Install]
WantedBy=multi-user.target
EOF
```

3. Legen Sie den Besitz der `sackd` Servicedatei fest.

```
sudo chown root:root /etc/systemd/system/sackd.service && \
    sudo chmod 0644 /etc/systemd/system/sackd.service
```

4. Aktivieren Sie den `sackd` Dienst.

```
sudo systemctl daemon-reload && sudo systemctl enable sackd
```

5. Starten Sie den Service `sackd`.

```
sudo systemctl start sackd
```

Schritt 6 — (Optional) Testen Sie die Verbindung

Vergewissern Sie sich, dass der `sackd` Dienst läuft. Beispiel für eine Ausgabe folgt. Wenn es Fehler gibt, werden sie normalerweise hier angezeigt.

```
[root@ip-10-3-27-112 ~]# systemctl status sackd
[x] sackd.service - Slurm auth and cred kiosk daemon
   Loaded: loaded (/etc/systemd/system/sackd.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2024-12-17 16:34:55 UTC; 8s ago
 Main PID: 9985 (sackd)
   CGroup: /system.slice/sackd.service
           ##9985 /opt/aws/pcs/scheduler/slurm-25.11/sbin/sackd --systemd --conf-
server=10.3.149.220:6817
```

```
Dec 17 16:34:55 ip-10-3-27-112.ec2.internal systemd[1]: Starting Slurm auth and cred
 kiosk daemon...
Dec 17 16:34:55 ip-10-3-27-112.ec2.internal systemd[1]: Started Slurm auth and cred
 kiosk daemon.
Dec 17 16:34:55 ip-10-3-27-112.ec2.internal sackd[9985]: sackd: running
```

Vergewissern Sie sich, dass die Verbindungen zum Cluster funktionieren, indem Sie Slurm-Client-Befehle wie `sinfo` und `squeue` verwenden. Hier ist eine Beispielausgabe von`sinfo`.

```
[root@ip-10-3-27-112 ~]# /opt/aws/pcs/scheduler/slurm-25.05/bin/sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
all up infinite 4 idle~ compute-[1-4]
```

Sie sollten auch Jobs einreichen können. Ein Befehl, der diesem Beispiel ähnelt, würde beispielsweise einen interaktiven Job auf einem Knoten im Cluster starten.

```
/opt/aws/pcs/scheduler/slurm-25.05/bin/srun --nodes=1 -p all --pty bash -i
```

Einen eigenständigen Anmeldeknoten mit mehreren Clustern in AWS PCS verbinden

Das `pcs-multi-cluster-login-configure.sh` Skript bietet eine automatisierte Möglichkeit, mehrere `sackd` Slurm-Daemons auf einem einzigen eigenständigen Login-Node zu konfigurieren. Es ermöglicht dem Login-Knoten, mit mehreren Clustern zu kommunizieren. Das Skript automatisiert die folgenden Operationen:

- Verwendet AWS PCS-API-Aktionen, um Clusterinformationen abzurufen
- Fordert zur Eingabe des Base64-codierten Slurm-Authentifizierungsschlüssels auf
- Erzeugt eine Slurm-JWKS-Datei mit Cluster-Authentifizierungsschlüssel
- Konfiguriert den `sackd` Dienst mit Cluster-Endpunkten und -Ports
- Erstellt eine `systemd` Servicedatei für einen clusterspezifischen Daemon `sackd`
- Generiert ein Aktivierungsskript für die Einrichtung der Clusterumgebung
- Aktiviert und startet den `sackd` Dienst

> **Note**
>
> Dieses Skript benötigt Slurm Version 25.05 oder höher.

> **Note**
>
> Für Slurm 25.11 oder höher können Sie `sackd --jwks-file <path>` und verwenden, `sackd --key-file <path>` um Authentifizierungsschlüsselpfade anstelle der Umgebungsvariablen anzugeben. `SLURM_SACK_JWKS` Der `SLURM_SACK_JWKS` Ansatz wird aus Gründen der Abwärtskompatibilität mit Slurm 25.05-Clustern weiterhin unterstützt.

Slurm muss bereits auf der Instanz installiert sein (entspricht Schritt 3 im manuellen Prozess). Die Instanz muss in der Lage sein, die Endpunkte des Zielclusters zu erreichen. Das Skript führt die entsprechenden Operationen aus Schritt 4 und Schritt 5 im manuellen Konfigurationsprozess aus. Es ruft automatisch die Clusterinformationen ab, konfiguriert den `sackd` Dienst, erstellt die erforderlichen `systemd` Dienstdateien und erstellt ein Aktivierungsskript, mit dem Benutzer ihre Shell-Umgebung für die Cluster-Interaktion konfigurieren können.

Topics

Voraussetzungen für das Konfigurationsskript für den AWS PCS-Multi-Cluster-Login-Knoten

Systemanforderungen

- Linux-Betriebssystem mit Unterstützung `systemd`
- Root-Rechte für die Systemkonfiguration

Erforderliche Befehle und Pakete

- `bash`— Shell-Interpreter (Version 4.0+)

- `curl`— Für den Abruf von AWS IMDS v2-Metadaten
- `jq`— JSON-Prozessor zum Analysieren von API-Antworten AWS
- `aws`— AWS CLI v2 zur Ausführung von AWS PCS-API-Aktionen und für den Secrets Manager Manager-Zugriff
- `systemctl`— `systemd` Servicemanagement
- `find`— Suchprogramm für das Dateisystem
- `grep`— Abgleich von Textmustern
- `sed`— Stream-Editor zur Textmanipulation
- `sort`— Hilfsprogramm zur Textsortierung
- `tail`— Zeigt die letzten Zeilen einer Datei an
- `mkdir`— Erstellung eines Verzeichnisses
- `chmod`— Ändert die Dateiberechtigungen
- `chown`— Ändert den Dateibesitz
- `ldconfig`— Dynamische Linker-Konfiguration

AWS Anforderungen

- Ein AWS PCS-Cluster, auf dem Slurm Version 25.05 oder höher ausgeführt wird
- AWS konfigurierte Anmeldeinformationen (über eine IAM-Rolle, eine Anmeldeinformationsdatei oder Umgebungsvariablen)
- Berechtigungen für:
  - `pcs:GetCluster`
  - `secretsmanager:GetSecretValue`(wenn Sie ein alternatives Geheimnis verwenden)

Systembenutzer und -gruppen

- Der `slurm` Benutzer und die Gruppe müssen auf dem System vorhanden sein

Slurm-Installation

- Slurm muss am selben Ort wie die AWS PCS Slurm-Installationspakete installiert werden:

```
/opt/aws/pcs/scheduler/slurm-version
```

AWS Skriptcode für die Konfiguration des PCS-Multi-Cluster-Anmeldeknotens

Speichern Sie den folgenden Quellcode in einer Datei mit dem folgenden Namen:

```
pcs-multi-cluster-login-configure.sh
```

Quellcode des Skripts

```
#!/bin/bash
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.

# AWS PCS Multi-Cluster Standalone Login Node Configuration Script
#
# This script configures AWS Parallel Computing Service (PCS) multi-cluster stand alone
 login nodes
# by setting up the Slurm authentication and credential kiosk daemon (sackd)
# for connecting to remote PCS clusters.
#
# Prerequisites:
# - AWS CLI configured with appropriate permissions
# - Slurm version 25.05 or later
# - Root privileges for system configuration
# - Network connectivity to AWS PCS endpoints


set -eo pipefail

# Function to display usage
usage() {
    echo "Usage: $0 --cluster-identifier <cluster-identifier> [--endpoint-url
 <endpoint-url>]"
    echo "       $0 -h|--help"
}

# Function to display help
help() {
    echo "AWS PCS Multi-Cluster Standalone Login Node Configuration Script"
```

```
    echo "================================================"
    echo
    echo "This script configures multi-cluster standalone login node for AWS Parallel
 Computing Service (PCS)"
    echo "by setting up the Slurm authentication and credential kiosk daemon (sackd)."
    echo
    usage
    echo
    echo "Options:"
    echo "  --cluster-identifier <id>    AWS PCS cluster identifier (required)"
    echo "  --endpoint-url <url>         Custom PCS endpoint URL (optional)"
    echo "  -h, --help                   Show this help message"
    echo
    echo "Examples:"
    echo "  $0 --cluster-identifier my-pcs-cluster"
    echo
    echo "Note: This script requires root privileges and Slurm version 25.05 or later."
}

# Function to retrieve authentication key
get_auth_key() {
    if [ "$ALTERNATE_SECRET_RETRIEVAL" = "true" ]; then
        echo "Retrieving authentication key from AWS Secrets Manager..." >&2
        local auth_key_arn=$(echo "$CLUSTER_INFO" | jq -r
 '.cluster.slurmConfiguration.authKey.secretArn')
        local auth_key_version=$(echo "$CLUSTER_INFO" | jq -r
 '.cluster.slurmConfiguration.authKey.secretVersion')

        if [ "$auth_key_arn" = "null" ] || [ "$auth_key_version" = "null" ]; then
            echo "Error: Auth key information not found in cluster configuration" >&2
            exit 1
        fi

        if ! aws secretsmanager get-secret-value --secret-id "$auth_key_arn" --version-
id "$auth_key_version" --query SecretString --output text --region "$REGION" 2>/dev/
null; then
            echo "Error: Failed to retrieve auth key from Secrets Manager" >&2
            exit 1
        fi
    else
        echo "Please enter the base64-encoded Slurm authentication key:" >&2
        echo -n "Base64 of the Slurm secret key: " >&2
        local key
        read -rs key
```

```
        echo >&2
        echo "$key"
    fi
}

# Function to get next available SACKD port
get_next_sackd_port() {
    local exclude_file="$1"
    local port=6918
    local used_ports=()

    # Get all currently used SACKD ports into an array
    while IFS= read -r line; do
        used_ports+=("$line")
    done < <(find /etc/sysconfig -name "sackd-pcs-*" ! -path "$exclude_file" \
             -exec grep SACKD_PORT= '{}' ';' 2>/dev/null | \
             sed 's/.*SACKD_PORT=//' | sort -n)

    # Loop through used ports to find first available port
    for used_port in "${used_ports[@]}"; do
        if [ "$port" -lt "$used_port" ]; then
            break
        elif [ "$port" -eq "$used_port" ]; then
            ((port++))
        fi
    done

    echo "$port"
}

# Function to configure cluster
configure_cluster() {
    mkdir -p /etc/slurm
    SLURM_JWKS_FILE="/etc/slurm/slurm-${CLUSTER_NAME}.jwks"
    echo '{"keys":
[{"alg":"HS256","kty":"oct","kid":"key-'"${CLUSTER_ID}"'","k":"'"${BASE64_SLURM_KEY}"'"}]}'
 | jq -c '.' > "${SLURM_JWKS_FILE}"

    chmod 0600 "$SLURM_JWKS_FILE"
    chown slurm:slurm "$SLURM_JWKS_FILE"

    SLURM_INSTALL_PATH="/opt/aws/pcs/scheduler/slurm-${SLURM_VERSION}"

    SACKD_RUNTIME_DIRECTORY="/run/slurm-${CLUSTER_NAME}"
```

```
    mkdir -p "${SACKD_RUNTIME_DIRECTORY}"
    chown slurm:slurm "${SACKD_RUNTIME_DIRECTORY}"

    mkdir -p /etc/sysconfig
    SACKD_SERVICE_NAME="sackd-pcs-${CLUSTER_NAME}"
    SACKD_SERVICE_ENV="/etc/sysconfig/${SACKD_SERVICE_NAME}"
    SACKD_PORT=$(get_next_sackd_port "$SACKD_SERVICE_ENV")
    cat > "${SACKD_SERVICE_ENV}" << EOF
SACKD_OPTIONS='--conf-server=$ENDPOINTS'
SLURM_SACK_JWKS='$SLURM_JWKS_FILE'
RUNTIME_DIRECTORY='$SACKD_RUNTIME_DIRECTORY'
SACKD_PORT=$SACKD_PORT
EOF

    SACKD_SERVICE_PATH="/etc/systemd/system/${SACKD_SERVICE_NAME}.service"

    cat << EOF > "$SACKD_SERVICE_PATH"
[Unit]
Description=Slurm auth and cred kiosk daemon
After=network-online.target remote-fs.target
Wants=network-online.target
ConditionPathExists=${SACKD_SERVICE_ENV}

[Service]
Type=notify
EnvironmentFile=${SACKD_SERVICE_ENV}
User=slurm
Group=slurm
RuntimeDirectory=slurm-${CLUSTER_NAME}
RuntimeDirectoryMode=0755
ExecStart=${SLURM_INSTALL_PATH}/sbin/sackd --systemd \$SACKD_OPTIONS
ExecReload=/bin/kill -HUP \$MAINPID
KillMode=process
LimitNOFILE=131072
LimitMEMLOCK=infinity
LimitSTACK=infinity

[Install]
WantedBy=multi-user.target
EOF

    chown root:root "$SACKD_SERVICE_PATH"
    chmod 0644 "$SACKD_SERVICE_PATH"
    systemctl daemon-reload && systemctl enable "$SACKD_SERVICE_NAME"
```

```
    systemctl restart "$SACKD_SERVICE_NAME"

    ACTIVATE_SCRIPT="activate-pcs-${CLUSTER_NAME}"
    cat > "$ACTIVATE_SCRIPT" << EOF
# Activate script for Slurm cluster ${CLUSTER_NAME}

# Add Slurm paths
export PATH="${SLURM_INSTALL_PATH}/bin:\$PATH"
export MANPATH="${SLURM_INSTALL_PATH}/share/man:\$MANPATH"
export LD_LIBRARY_PATH="${SLURM_INSTALL_PATH}/lib:\$LD_LIBRARY_PATH"
ldconfig

# Set Slurm configuration
export SLURM_CONF="/run/slurm-${CLUSTER_NAME}/conf/slurm.conf"
export PCS_CLUSTER_NAME="${CLUSTER_NAME}"
export PCS_CLUSTER_IDENTIFIER="${CLUSTER_IDENTIFIER}"
export PCS_CLUSTER_ID="${CLUSTER_ID}"

echo "Activated PCS cluster environment: ${CLUSTER_NAME}"

# Deactivate function
function deactivate-pcs-${CLUSTER_NAME}() {
    export PATH="\$(echo "\$PATH" | sed -e "s|${SLURM_INSTALL_PATH}/bin:||g" -e "s|:
${SLURM_INSTALL_PATH}/bin||g" -e "s|^${SLURM_INSTALL_PATH}/bin\$||")"
    export MANPATH="\$(echo "\$MANPATH" | sed -e "s|${SLURM_INSTALL_PATH}/share/man:||
g" -e "s|:${SLURM_INSTALL_PATH}/share/man||g" -e "s|^${SLURM_INSTALL_PATH}/share/man\
$||")"
    export LD_LIBRARY_PATH="\$(echo "\$LD_LIBRARY_PATH" | sed -e "s|
${SLURM_INSTALL_PATH}/lib:||g" -e "s|:${SLURM_INSTALL_PATH}/lib||g" -e "s|^
${SLURM_INSTALL_PATH}/lib\$||")"
    unset SLURM_CONF
    unset PCS_CLUSTER_NAME
    unset PCS_CLUSTER_IDENTIFIER
    unset PCS_CLUSTER_ID
    unset -f deactivate-pcs-${CLUSTER_NAME}
    ldconfig
    echo "Deactivated PCS cluster environment: ${CLUSTER_NAME}"
}

export -f deactivate-pcs-${CLUSTER_NAME}

EOF
}
```

```
# Main function
main() {
    # Parse arguments
    CLUSTER_IDENTIFIER=""
    PCS_ENDPOINT_URL=""

    while [ "$1" != "" ]; do
        case $1 in
            --cluster-identifier)
                shift
                CLUSTER_IDENTIFIER="$1"
                ;;
            --endpoint-url)
                shift
                PCS_ENDPOINT_URL="--endpoint-url $1"
                ;;
            -h|--help)
                help
                exit 0
                ;;
            *)
                echo "Invalid argument: $1" >&2
                usage >&2
                exit 1
                ;;
        esac
        shift
    done

    # Validate required arguments
    if [ -z "$CLUSTER_IDENTIFIER" ]; then
        echo "Error: --cluster-identifier is required" >&2
        usage >&2
        exit 1
    fi

    # Validate running as root
    if [ "$EUID" -ne 0 ]; then
        echo "Error: This script must be run as root" >&2
        exit 1
    fi

    # Validate required commands are available
    for cmd in aws jq curl; do
```

```
        if ! command -v "$cmd" &> /dev/null; then
            echo "Error: Required command '$cmd' not found" >&2
            exit 1
        fi
    done

    # Get the region name from IMDS v2 with error handling (try IPv6 first, fallback to
 IPv4)
    echo "Retrieving AWS region from instance metadata..."
    # Try IPv6 IMDS endpoint first (fd00:ec2::254) with fast timeout (1s connect, 2s
 total)
    # If IPv6 fails, fallback to IPv4 IMDS endpoint (169.254.169.254)
    IMDS_ENDPOINT="http://[fd00:ec2::254]"
    if ! TOKEN=$(curl -s -X PUT "${IMDS_ENDPOINT}/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600" --connect-timeout 1 --max-time 2 2>/dev/null); then
        IMDS_ENDPOINT="http://169.254.169.254"
        if ! TOKEN=$(curl -s -X PUT "${IMDS_ENDPOINT}/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600" --max-time 5); then
            echo "Error: Failed to retrieve IMDS token. Ensure this script is running
 on an EC2 instance." >&2
            exit 1
        fi
    fi

    if ! REGION=$(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" "${IMDS_ENDPOINT}/
latest/dynamic/instance-identity/document" --max-time 5 | jq -r '.region'); then
        echo "Error: Failed to retrieve AWS region from instance metadata" >&2
        exit 1
    fi

    echo "Detected AWS region: $REGION"

    # Retrieve cluster information from AWS PCS
    echo "Retrieving cluster information for: $CLUSTER_IDENTIFIER"
    # shellcheck disable=SC2086
    if ! CLUSTER_INFO=$(aws pcs get-cluster --region "$REGION" --cluster-identifier
 "$CLUSTER_IDENTIFIER" $PCS_ENDPOINT_URL 2>/dev/null); then
        echo "Error: Failed to retrieve cluster information. Check cluster identifier
 and AWS permissions." >&2
        exit 1
    fi

    CLUSTER_ID=$(echo "$CLUSTER_INFO" | jq -r '.cluster.id')
    CLUSTER_NAME="$(echo "$CLUSTER_INFO" | jq -r '.cluster.name')"
```

```
    SLURM_VERSION=$(echo "$CLUSTER_INFO" | jq -r '.cluster.scheduler.version')
    SLURM_VERSION=${SLURM_VERSION#Slurm_}

    # Check if Slurm version is >= 25.05
    # shellcheck disable=SC2072
    if [[ "$SLURM_VERSION" < "25.05" ]]; then
        echo "Error: This script requires Slurm version 25.05 or later. Found version:
$SLURM_VERSION" >&2
        exit 1
    fi

    ENDPOINTS=$(echo "$CLUSTER_INFO" | jq -r '.cluster.endpoints[] | select(.type
== "SLURMCTLD") | (if .privateIpAddress != "" then .privateIpAddress else "["
+ .ipv6Address + "]" end) + ":" + .port' | tr '\n' ',' | sed 's/,$//')

    # Get BASE64_SLURM_KEY
    BASE64_SLURM_KEY=$(get_auth_key)

    if [ -z "$BASE64_SLURM_KEY" ]; then
        echo "Error: base64 Slurm key cannot be empty" >&2
        exit 1
    fi

    configure_cluster

    # Final configuration summary
    echo "========================================="
    echo "Configuration completed successfully!"
    echo "========================================="
    echo "Cluster Name: $CLUSTER_NAME"
    echo "Cluster ID: $CLUSTER_ID"
    echo "Slurm Version: $SLURM_VERSION"
    echo "Service Name: $SACKD_SERVICE_NAME"
    echo "SACKD Port: $SACKD_PORT"
    echo
    echo "To activate this cluster environment, run:"
    echo "  source ./$ACTIVATE_SCRIPT"
    echo
    echo "To deactivate this cluster environment, run:"
    echo "  deactivate-pcs-${CLUSTER_NAME}"
    echo
    echo "To check service status:"
    echo "  systemctl status $SACKD_SERVICE_NAME"
    echo
```

```
    echo "To view service logs:"
    echo "  journalctl -u $SACKD_SERVICE_NAME -f"
}

# Exit if being sourced for testing
[[ "${BASH_SOURCE[0]}" != "${0}" ]] && return

# Execute main function
main "$@"
```

Verwenden des Konfigurationsskripts für den AWS PCS-Multi-Cluster-Anmeldeknoten

Ausführen des Skripts

So führen Sie das Konfigurationsskript aus:

1. Speichern Sie den Inhalt des Skripts in einer Datei mit dem Namen:

   ```
   pcs-multi-cluster-login-configure.sh
   ```

2. Machen Sie es ausführbar:

   ```
   chmod +x pcs-multi-cluster-login-configure.sh
   ```

3. Führen Sie das Skript aus:

   ```
   ./pcs-multi-cluster-login-configure.sh --cluster-identifier cluster-name
   ```

Umgebungen für Cluster-Interaktionen

Nach erfolgreicher Konfiguration generiert das Skript ein clusterspezifisches Aktivierungsskript im aktuellen Verzeichnis. Das Skript hat den Namen. `activate-pcs-`*`cluster-name`* Das Aktivierungsskript konfiguriert die erforderlichen Umgebungsvariablen und Pfade für die Interaktion mit dem Zielcluster.

Um eine Clusterumgebung zu aktivieren

- Verwenden Sie den `source` Befehl, um das Aktivierungsskript auszuführen

```
source ./activate-pcs-cluster-name
```

Example

```
# Activate cluster environment for cluster 'my-cluster'
source ./activate-pcs-my-cluster

# Now you can use Slurm commands
sinfo
squeue
sbatch my-job.sh
```

Was macht das Aktivierungsskript

- Legt die SLURM_CONF Umgebungsvariable so fest, dass sie auf die Konfiguration des Clusters verweist.
- Aktualisiert denPATH, sodass er die Slurm-Binärdateien des Clusters enthält.
- Konfiguriert andere notwendige Slurm-Umgebungsvariablen (,). MANPATH LD_LIBRARY_PATH
- Legt Variablen zur Identifizierung des AWS PCS-Clusters fest.
- Ermöglicht eine nahtlose Interaktion mit dem AWS PCS-Zielcluster.

Um eine Cluster-Umgebung zu deaktivieren

- Führen Sie den Deaktivierungsbefehl aus.

```
deactivate-pcs-cluster-name
```

Example

```
# After activating a cluster
source ./activate-pcs-my-cluster

# Work with the cluster
sinfo

# Deactivate when done
```

```
deactivate-pcs-my-cluster
```

Was macht der Deaktivierungsbefehl

- Stellt die ursprüngliche PATH Umgebungsvariable wieder her.
- Setzt clusterspezifische Slurm-Umgebungsvariablen zurück.
- Setzt die Shell-Umgebung in ihren Zustand vor der Aktivierung zurück.

> **Note**
>
> Die Aktivierung ist sitzungsspezifisch und muss in der Shell-Sitzung erfolgen, in der Sie mit dem Cluster interagieren möchten.

AWS PCS-Netzwerke

Ihr AWS PCS-Cluster wird in einer Amazon VPC erstellt. Dieses Kapitel enthält die folgenden Themen über Netzwerke für den Scheduler und die Knoten Ihres Clusters.

Abgesehen von der Auswahl eines Subnetzes, in dem Instances gestartet werden sollen, müssen Sie EC2 Startvorlagen verwenden, um das Netzwerk für AWS PCS-Compute-Knotengruppen zu konfigurieren. Weitere Informationen über Startvorlagen finden Sie unter Verwenden von Amazon EC2 EC2-Startvorlagen mit AWS PCS.

Themen

AWS Anforderungen und Überlegungen zu PCS VPC und Subnetzen

Wenn Sie einen AWS PCS-Cluster erstellen, geben Sie eine VPC als Subnetz in dieser VPC an. Dieses Thema bietet einen Überblick über die AWS PCS-spezifischen Anforderungen und Überlegungen für die VPC und die Subnetze, die Sie mit Ihrem Cluster verwenden. Wenn Sie keine VPC haben, die Sie mit AWS PCS verwenden können, können Sie eine mit einer AWS bereitgestellten CloudFormation Vorlage erstellen. Weitere Informationen finden Sie VPCs unter Virtual Private Clouds (VPC) im Amazon VPC-Benutzerhandbuch.

VPC-Anforderungen und -Überlegungen

Wenn Sie einen Cluster erstellen, muss die von Ihnen angegebene VPC die folgenden Anforderungen und Überlegungen erfüllen:

- Die VPC muss über eine ausreichende Anzahl von IP-Adressen für den Cluster, alle Knoten und andere Clusterressourcen verfügen, die Sie erstellen möchten. Weitere Informationen finden Sie unter IP-Adressierung für Ihre VPCs und Subnetze im Amazon VPC-Benutzerhandbuch.
- Wenn Ihr Cluster Folgendes verwendet: IPv6
  - Ordnen Sie Ihrer VPC einen IPv6 CIDR-Block zu. Weitere Informationen finden Sie unter VPC erstellen im Amazon-VPC-Benutzerhandbuch.

    > **Important**
    >
    > Sie können Ihre VPC zwar IPv4 sowohl mit als auch konfigurieren IPv6, aber Sie können nur einen Netzwerktyp für Ihren Cluster auswählen.

  - Aktivieren Sie die automatische IPv6 Adresszuweisung für Ihre Subnetze.
  - Weitere Informationen finden Sie unter:
    - IPv6 ein AWS
    - Die IPv6 Adressierung auf AWS verstehen und einen skalierbaren Adressierungsplan entwerfen
- Die VPC muss über einen DNS-Hostnamen und eine Unterstützung für DNS-Auflösung verfügen. Andernfalls können Knoten den Kundencluster nicht registrieren. Weitere Infomationen finden Sie unter DNS-Attribute für Ihre VPC im Amazon VPC-Benutzerhandbuch.
- Für die VPC müssen möglicherweise VPC-Endpunkte verwendet werden AWS PrivateLink , um die PCS-API kontaktieren zu können. AWS Weitere Informationen finden Sie unter Connect Ihrer VPC mit Services AWS PrivateLink im Amazon VPC-Benutzerhandbuch.

> **Important**
>
> AWS PCS unterstützt keine VPC mit dedizierter Instance-Tenancy. Die VPC, die Sie für AWS PCS verwenden, muss `default` Instance-Tenancy verwenden. Sie können die Instance-Tenancy für eine bestehende VPC ändern. Weitere Informationen finden Sie unter Ändern der Instance-Tenancy einer VPC im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Subnetz-Anforderungen und -Überlegungen

Wenn Sie einen Slurm-Cluster erstellen, erstellt AWS PCS ein Elastic Network Interface (ENI) in dem von Ihnen angegebenen Subnetz. Diese Netzwerkschnittstelle ermöglicht die Kommunikation zwischen dem Scheduler-Controller und der Kunden-VPC. Die Netzwerkschnittstelle ermöglicht es Slurm auch, mit den in Ihrem Konto bereitgestellten Komponenten zu kommunizieren. Sie können das Subnetz für einen Cluster nur zum Zeitpunkt der Erstellung angeben.

Subnetzanforderungen für Cluster

Das Subnetz, das Sie bei der Erstellung eines Clusters angeben, muss die folgenden Anforderungen erfüllen:

- Das Subnetz muss mindestens eine IP-Adresse haben, damit es von PCS verwendet werden AWS kann.
- Wenn Ihr Cluster verwendet IPv6, müssen alle Subnetze in Ihrem Cluster verwenden. IPv6

> **Important**
>
> Compute-Knotengruppen, die mit AWS PCS Sample AMIs und mehreren Netzwerkschnittstellen konfiguriert sind, funktionieren derzeit nicht, wenn die Subnetze nur für die Verwendung konfiguriert sind. IPv6 Verwenden Sie stattdessen Dual-Stack-Subnetze (IPv4 und IPv6) oder IPv4 reine Subnetze. Weitere Informationen finden Sie unter Verwenden von Amazon Machine Images (AMIs) -Beispieldateien mit AWS STK..

- Das Subnetz darf sich nicht in AWS Outposts, AWS Wavelength oder einer lokalen Zone befinden. AWS
- Das Subnetz kann öffentlich oder privat sein. Wir empfehlen, dass Sie, wenn möglich, ein privates Subnetz angeben. Ein öffentliches Subnetz ist ein Subnetz mit einer Routing-Tabelle, die eine Route zu einem Internet-Gateway enthält. Ein privates Subnetz ist ein Subnetz mit einer Routing-Tabelle, das keine Route zu einem Internet-Gateway enthält.

Subnetzanforderungen für Knoten

Sie können Knoten und andere Clusterressourcen in dem Subnetz bereitstellen, das Sie bei der Erstellung Ihres AWS PCS-Clusters angeben, sowie in anderen Subnetzen in derselben VPC.

Jedes Subnetz, in dem Sie Knoten und Clusterressourcen bereitstellen, muss die folgenden Anforderungen erfüllen:

- Sie müssen sicherstellen, dass das Subnetz über genügend verfügbare IP-Adressen verfügt, um alle Knoten und Clusterressourcen bereitzustellen.
- Wenn Ihr Cluster Knoten verwendet IPv4 und Sie planen, Knoten in einem öffentlichen Subnetz bereitzustellen, muss dieses Subnetz automatisch öffentliche Adressen zuweisen IPv4 .

> **Note**
>
> Instances in einem öffentlichen Subnetz müssen eine Sicherheitsgruppe mit Regeln für eingehenden Datenverkehr verwenden, die Datenverkehr von öffentlichen IP-Adressen zulassen. Sofern Sie keine spezifischen Einschränkungen für Quelladressen haben, bedeutet dies eine IPv4 Quelladresse von 0.0.0.0/0 oder eine IPv6 Quelladresse von: :/0.

- Wenn es sich bei dem Subnetz, in dem Sie Knoten bereitstellen, um ein privates Subnetz handelt und die Routing-Tabelle keine Route zu einem NAT-Gerät (Network Address Translation) () enthält, fügen Sie der IPv4 Kunden-VPC VPC-Endpunkte AWS PrivateLink hinzu, die dies verwenden. VPC-Endpunkte werden für alle AWS Dienste benötigt, mit denen die Knoten Kontakt aufnehmen. Der einzige erforderliche Endpunkt besteht darin, dass AWS PCS es dem Knoten ermöglicht, die `RegisterComputeNodeGroupInstance` API-Aktion aufzurufen. Weitere Informationen finden Sie RegisterComputeNodeGroupInstancein der AWS PCS-API-Referenz.
- Der Status eines öffentlichen oder privaten Subnetzes hat keinen Einfluss auf AWS PCS. Die erforderlichen Endpunkte müssen erreichbar sein.

Erstellen Sie eine VPC für Ihr AWS PCS-Cluster

Sie können innerhalb von AWS Parallel Computing Service (PCS) eine Amazon Virtual Private Cloud (Amazon AWS VPC) für Ihre Cluster erstellen.

Verwenden Sie Amazon VPC, um VPC-Ressourcen in einem von Ihnen definierten virtuellen Netzwerk zu starten. Dieses virtuelle Netzwerk ist einem herkömmlichen Netzwerk, das Sie in Ihrem eigenen Rechenzentrum betreiben, sehr ähnlich. Es bietet jedoch die Vorzüge, die mit der Nutzung der skalierbaren Infrastruktur von Amazon Web Services einhergehen. Wir empfehlen, dass Sie sich mit dem Amazon VPC-Service gründlich auskennen, bevor Sie VPC-Produktionscluster bereitstellen. Weitere Informationen finden Sie unter Was ist Amazon VPC? im visuellen Autorenmodus. Amazon VPC-Benutzerhandbuch.

Ein PCS-Cluster, Knoten und unterstützende Ressourcen (wie Dateisysteme und Verzeichnisdienste) werden in Ihrer Amazon VPC bereitgestellt. Wenn Sie eine bestehende Amazon VPC mit PCS verwenden möchten, muss sie die unter beschriebenen Anforderungen erfüllen. AWS Anforderungen und Überlegungen zu PCS VPC und Subnetzen In diesem Thema wird beschrieben, wie Sie mithilfe einer AWS bereitgestellten CloudFormation Vorlage eine VPC erstellen, die die PCS-Anforderungen erfüllt. Sobald Sie eine Vorlage bereitgestellt haben, können Sie sich die mit der Vorlage erstellten Ressourcen ansehen, um genau zu erfahren, welche Ressourcen sie erstellt hat und wie diese Ressourcen konfiguriert sind.

Voraussetzungen

Um eine Amazon VPC für PCS zu erstellen, benötigen Sie die erforderlichen IAM-Berechtigungen, um Amazon VPC-Ressourcen zu erstellen. Diese Ressourcen sind VPCs, Subnetze, Sicherheitsgruppen, Routing-Tabellen und Routen sowie Internet- und NAT-Gateways. Weitere Informationen finden Sie unter Erstellen einer VPC mit einem öffentlichen Subnetz im Amazon VPC-Benutzerhandbuch. Die vollständige Liste für Amazon EC2 finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon EC2 in der Service Authorization Reference.

Erstellen Sie eine Amazon VPC

Erstellen Sie eine VPC, indem Sie die entsprechende URL für den Ort, an AWS-Region dem Sie PCS verwenden möchten, kopieren und einfügen. Sie können die CloudFormation Vorlage auch herunterladen und selbst auf die CloudFormation Konsole hochladen.

- USA Ost (Nord-Virginia) (us-east-1)

```
https://console.aws.amazon.com/cloudformation/home?region=us-east-1#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- USA Ost (Ohio) (us-east-2)

```
https://console.aws.amazon.com/cloudformation/home?region=us-east-2#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- USA West (Oregon) (us-west-2)

```
https://console.aws.amazon.com/cloudformation/home?region=us-west-2#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Asien-Pazifik (Mumbai) (ap-south-1)

```
https://console.aws.amazon.com/cloudformation/home?region=ap-south-1#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Asien-Pazifik (Singapur) (ap-southeast-1)

```
https://console.aws.amazon.com/cloudformation/home?region=ap-southeast-1#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Asien-Pazifik (Sydney) (ap-southeast-2)

```
https://console.aws.amazon.com/cloudformation/home?region=ap-southeast-2#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Asien-Pazifik (Tokio) (ap-northeast-1)

```
https://console.aws.amazon.com/cloudformation/home?region=ap-northeast-1#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Asien-Pazifik (Osaka) (ap-northeast-3)

```
https://console.aws.amazon.com/cloudformation/home?region=ap-northeast-3#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Europa (Frankfurt) (eu-central-1)

```
https://console.aws.amazon.com/cloudformation/home?region=eu-central-1#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Europa (Irland) (eu-west-1)

```
https://console.aws.amazon.com/cloudformation/home?region=eu-west-1#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Europa (London) (eu-west-2)

```
https://console.aws.amazon.com/cloudformation/home?region=eu-west-2#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Europa (Paris) (eu-west-3)

```
https://console.aws.amazon.com/cloudformation/home?region=eu-west-3#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Europa (Mailand) (eu-south-1)

```
https://console.aws.amazon.com/cloudformation/home?region=eu-south-1#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Europa (Spanien) (eu-south-2)

```
https://console.aws.amazon.com/cloudformation/home?region=eu-south-2#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Europa (Stockholm) (eu-north-1)

```
https://console.aws.amazon.com/cloudformation/home?region=eu-north-1#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Südamerika (São Paulo) (sa-east-1)

```
https://console.aws.amazon.com/cloudformation/home?region=sa-east-1#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- AWS GovCloud (US-East) (us-gov-east-1)

```
https://console.aws.amazon.com/cloudformation/home?region=us-gov-east-1#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- AWS GovCloud () (US-Regierung West-1) US-West

```
https://console.aws.amazon.com/cloudformation/home?region=us-gov-west-1#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Nur Vorlage

```
https://aws-hpc-recipes.s3.us-east-1.amazonaws.com/main/recipes/net/hpc_large_scale/
assets/main.yaml
```

So erstellen Sie eine Amazon VPC für PCS

1. Öffnen Sie die Vorlage in der CloudFormation Konsole.

   > **Note**
   >
   > Diese Werte sind in der Vorlage bereits ausgefüllt, sodass Sie sie einfach als Standardwerte beibehalten können.

2. Geben Sie unter Geben Sie einen Stacknamen ein und dann Stackname die folgenden Werte ein`hpc-networking`.
3. Geben Sie unter Parameter die folgenden Details ein:

   a. Geben Sie dann unter VPC CidrBlockein `10.3.0.0/16`

   b. Unter Subnetze A:

      i. Geben Sie dann CidrPublicSubnetAFolgendes ein `10.3.0.0/20`

      ii. Geben Sie CidrPrivateSubnetAdann ein `10.3.128.0/20`

   c. Unter Subnetze B:

      i. Geben Sie dann CidrPublicSubnetBFolgendes ein `10.3.16.0/20`

      ii. Geben Sie CidrPrivateSubnetAdann ein `10.3.144.0/20`

d. Unter Subnetze C:

i. Wählen Sie für ProvisionSubnetsC die Option aus`True`.

> **Note**
>
> Wenn Sie eine VPC in einer Region mit weniger als drei Availability Zones erstellen, wird diese Option ignoriert, wenn sie auf `True` gesetzt ist.

ii. Geben Sie dann CidrPublicSubnetBFolgendes ein `10.3.32.0/20`

iii. Geben Sie CidrPrivateSubnetAdann ein `10.3.160.0/20`

4. Aktivieren Sie unter Funktionen das Kontrollkästchen Ich bestätige, dass AWS CloudFormation möglicherweise IAM-Ressourcen erstellt.

Überwachen Sie den Status des CloudFormation Stacks. Wenn es erreicht ist`CREATE_COMPLETE`, sind die VPC-Ressourcen für Sie einsatzbereit.

> **Note**
>
> Um alle Ressourcen zu sehen, die mit der CloudFormation Vorlage erstellt wurden, öffnen Sie die CloudFormation Konsole. Wählen Sie das `hpc-networking`-Stack, und wählen Sie dann die Registerkarte Ressourcen.

Sicherheitsgruppen in AWS PCS

Sicherheitsgruppen in Amazon EC2 dienen als virtuelle Firewalls zur Steuerung des ein- und ausgehenden Datenverkehrs zu Instances. Verwenden Sie eine Startvorlage für eine AWS PCS-Compute-Knotengruppe, um Sicherheitsgruppen zu ihren Instances hinzuzufügen oder zu entfernen. Wenn Ihre Startvorlage keine Netzwerkschnittstellen enthält, verwenden Sie diese, `SecurityGroupIds` um eine Liste von Sicherheitsgruppen bereitzustellen. Wenn Ihre Startvorlage Netzwerkschnittstellen definiert, müssen Sie den `Groups` Parameter verwenden, um jeder Netzwerkschnittstelle Sicherheitsgruppen zuzuweisen. Weitere Informationen über Startvorlagen finden Sie unter Verwenden von Amazon EC2 EC2-Startvorlagen mit AWS PCS.

> **Note**
>
> Änderungen an der Sicherheitsgruppenkonfiguration in der Startvorlage wirken sich nur auf neue Instances aus, die nach der Aktualisierung der Compute-Knotengruppe gestartet werden.

Anforderungen und Überlegungen zur Sicherheitsgruppe

AWS PCS erstellt ein kontenübergreifendes Elastic Network Interface (ENI) in dem Subnetz, das Sie bei der Erstellung eines Clusters angeben. Dies bietet dem HPC-Scheduler, der in einem von PCS verwalteten Konto ausgeführt wird AWS, einen Pfad für die Kommunikation mit EC2-Instances, die von PCS gestartet wurden. AWS Sie müssen eine Sicherheitsgruppe für diese ENI bereitstellen, die eine bidirektionale Kommunikation zwischen dem Scheduler-ENI und Ihren Cluster-EC2-Instances ermöglicht.

Eine einfache Möglichkeit, dies zu erreichen, besteht darin, eine permissive, selbstreferenzierende Sicherheitsgruppe zu erstellen, die den TCP/IP Datenverkehr auf allen Ports zwischen allen Mitgliedern der Gruppe zulässt. Sie können dies sowohl dem Cluster als auch den EC2-Instances der Knotengruppe zuordnen.

Beispiel für eine permissive Sicherheitsgruppenkonfiguration

IPv4

Art der Regel	Protokolle	Ports	Quelle	Ziel
Eingehend	Alle	Alle	Selbst	
Ausgehend	Alle	Alle		0.0.0.0/0
Ausgehend	Alle	Alle		Selbst

IPv6

Art der Regel	Protokolle	Ports	Quelle	Ziel
Eingehend	Alle	Alle	Selbst	

Art der Regel	Protokolle	Ports	Quelle	Ziel
Ausgehend	Alle	Alle		::/0
Ausgehend	Alle	Alle		Selbst

Diese Regeln ermöglichen den freien Fluss des gesamten Datenverkehrs zwischen dem Slurm-Controller und den Knoten, lassen den gesamten ausgehenden Verkehr zu einem beliebigen Ziel zu und ermöglichen EFA-Verkehr.

Beispiel für eine restriktive Sicherheitsgruppenkonfiguration

Sie können auch die offenen Ports zwischen dem Cluster und seinen Rechenknoten einschränken. Für den Slurm-Scheduler muss die mit Ihrem Cluster verbundene Sicherheitsgruppe die folgenden Ports zulassen:

- 6817 — aktiviert eingehende Verbindungen zu EC2-Instances `slurmctld`
- 6818 — ermöglicht ausgehende Verbindungen von `slurmctld` und zur Ausführung auf EC2-Instances `slurmd`

Die mit Ihren Rechenknoten verbundene Sicherheitsgruppe muss die folgenden Ports zulassen:

- 6817 — ermöglicht ausgehende Verbindungen zu `slurmctld` EC2-Instances.
- 6818 — ermöglicht eingehende und ausgehende Verbindungen `slurmd` von `slurmctld` und zu Knotengruppen-Instances `slurmd`
- 60001—63000 — Unterstützung eingehender und ausgehender Verbindungen zwischen Knotengruppen-Instances `srun`
- EFA-Verkehr zwischen Knotengruppen-Instances. Weitere Informationen finden Sie unter Vorbereiten einer EFA-fähigen Sicherheitsgruppe im Benutzerhandbuch für Linux-Instances
- Jeder andere Datenverkehr zwischen den Knoten, der für Ihren Workload erforderlich ist

Mehrere Netzwerkschnittstellen in AWS PCS

Einige EC2-Instances haben mehrere Netzwerkkarten. Dadurch können sie eine höhere Netzwerkleistung bieten, einschließlich Bandbreitenkapazitäten von über 100 Gbit/s und verbesserter

Paketverarbeitung. Weitere Informationen zu Instances mit mehreren Netzwerkkarten finden Sie unter Elastic Network Interfaces im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Konfigurieren Sie zusätzliche Netzwerkkarten für Instances in einer AWS PCS-Compute-Knotengruppe, indem Sie der zugehörigen EC2-Startvorlage Netzwerkschnittstellen hinzufügen. Im Folgenden finden Sie ein Beispiel für eine Startvorlage, die zwei Netzwerkkarten aktiviert, wie sie sich beispielsweise auf einer `hpc7a.96xlarge` Instance befinden. Beachten Sie folgende Details:

- Das Subnetz für jede Netzwerkschnittstelle muss das gleiche sein, das Sie bei der Konfiguration der AWS PCS-Compute-Knotengruppe ausgewählt haben, die die Startvorlage verwendet.
- Das primäre Netzwerkgerät, auf dem routinemäßige Netzwerkkommunikation wie SSH- und HTTPS-Verkehr stattfindet, wird durch die Einstellung von eingerichtet. `DeviceIndex` `0` Andere Netzwerkschnittstellen haben einen Wert `DeviceIndex` von. `1` Es kann nur eine primäre Netzwerkschnittstelle geben — alle anderen Schnittstellen sind sekundär.
- Alle Netzwerkschnittstellen müssen eindeutig sein. `NetworkCardIndex` Es wird empfohlen, sie sequenziell zu nummerieren, so wie sie in der Startvorlage definiert sind.
- Sicherheitsgruppen für jede Netzwerkschnittstelle werden mithilfe von `Groups` festgelegt. In diesem Beispiel wird der primären Netzwerkschnittstelle eine eingehende SSH-Sicherheitsgruppe (`sg-`*`SshSecurityGroupId`*) hinzugefügt, ebenso wie die Sicherheitsgruppe, die die Kommunikation innerhalb des Clusters ermöglicht (). `sg-`*`ClusterSecurityGroupId`* Schließlich wird sowohl der primären als auch der sekundären Schnittstelle eine Sicherheitsgruppe hinzugefügt, die ausgehende Verbindungen zum Internet (`sg-`*`InternetOutboundSecurityGroupId`*) ermöglicht.

```
{
    "NetworkInterfaces": [
        {
            "DeviceIndex": 0,
            "NetworkCardIndex": 0,
            "SubnetId": "subnet-SubnetId",
            "Groups": [
               "sg-SshSecurityGroupId",
               "sg-ClusterSecurityGroupId",
               "sg-InternetOutboundSecurityGroupId"
            ]
        },
        {
            "DeviceIndex": 1,
```

```
            "NetworkCardIndex": 1,
            "SubnetId": "subnet-SubnetId",
            "Groups": ["sg-InternetOutboundSecurityGroupId"]
        }
    ]
}
```

Platzierungsgruppen für EC2-Instances in AWS PCS

Sie können eine Platzierungsgruppe verwenden, um die Platzierung von EC2-Instances so zu beeinflussen, dass sie den Anforderungen der Arbeitslast entspricht, die auf ihnen ausgeführt wird.

Typen von Platzierungsgruppen

- Cluster — Fügt Instanzen nahe beieinander in einer Availability Zone zusammen, um die Kommunikation mit niedriger Latenz zu optimieren.
- Partition — Verteilt Instanzen auf logische Partitionen, um die Ausfallsicherheit zu maximieren.
- Verteilung — Erzwingt strikt, dass eine kleine Anzahl von Instances auf unterschiedlicher Hardware gestartet wird, was auch zur Erhöhung der Ausfallsicherheit beitragen kann.

Weitere Informationen finden Sie unter Platzierungsgruppen für Ihre Amazon EC2 EC2-Instances im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Wir empfehlen, eine Cluster-Platzierungsgruppe einzubeziehen, wenn Sie eine AWS PCS-Compute-Knotengruppe für die Verwendung des Elastic Fabric Adapter (EFA) konfigurieren.

Um eine Cluster-Platzierungsgruppe zu erstellen, die mit EFA funktioniert

1. Erstellen Sie eine Platzierungsgruppe mit dem Typ Cluster für die Compute-Knotengruppe.
   - Verwenden Sie den folgenden AWS CLI Befehl:

     ```
     aws ec2 create-placement-group --strategy cluster --group-name PLACEMENT-GROUP-NAME
     ```

   - Sie können auch eine CloudFormation Vorlage verwenden, um eine Platzierungsgruppe zu erstellen. Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch unter Arbeiten mit CloudFormation Vorlagen. Laden Sie die Vorlage von der folgenden URL herunter und laden Sie sie in die CloudFormation Konsole hoch.

```
https://aws-hpc-recipes.s3.amazonaws.com/main/recipes/pcs/enable_efa/assets/efa-placement-group.yaml
```

2. Nehmen Sie die Platzierungsgruppe in die EC2-Startvorlage für die AWS PCS-Compute-Knotengruppe auf.

Verwenden des Elastic Fabric Adapter (EFA) mit AWS PCS

Elastic Fabric Adapter (EFA) ist eine leistungsstarke, fortschrittliche Netzwerkverbindung AWS , die Sie an Ihre EC2-Instance anschließen können, um High Performance Computing (HPC) und Machine-Learning-Anwendungen zu beschleunigen. Um Ihre Anwendungen, die auf einem AWS PCS-Cluster mit EFA ausgeführt werden, zu aktivieren, müssen Sie die Instances der AWS PCS-Compute-Knotengruppe so konfigurieren, dass EFA wie folgt verwendet wird.

> **Note**
>
> EFA auf einem AWS PCS-compatible AMI installieren — Auf dem in der AWS PCS-Compute-Knotengruppe verwendeten AMI muss der EFA-Treiber installiert und geladen sein. Informationen zum Erstellen eines benutzerdefinierten AMI mit installierter EFA-Software finden Sie unterBenutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS.

Inhalt

Identifizieren Sie EFA-enabled EC2-Instances

Um EFA verwenden zu können, müssen alle Instance-Typen, die für eine AWS PCS-Compute-Gruppe zulässig sind, EFA unterstützen und dieselbe Anzahl von vCPUs (und GPUs, falls zutreffend) haben. Eine Liste der EFA-enabled Instances finden Sie unter Elastic Fabric Adapter für HPC- und ML-Workloads auf Amazon EC2 im Amazon Elastic Compute Cloud-Benutzerhandbuch. Sie können den auch verwenden AWS CLI , um eine Liste der Instance-Typen anzuzeigen, die EFA unterstützen. *region-code*Ersetzen Sie es durch den AWS-Region Ort, an dem Sie AWS PCS verwenden, z. B. `us-east-1`

```
aws ec2 describe-instance-types \
   --region region-code \
   --filters Name=network-info.efa-supported,Values=true \
   --query "InstanceTypes[*].[InstanceType]" \
   --output text | sort
```

> **Note**
>
> Ermitteln Sie, wie viele Netzwerkschnittstellen verfügbar sind — Einige EC2-Instances verfügen über mehrere Netzwerkkarten. Dadurch können sie über mehrere EFAs verfügen. Weitere Informationen finden Sie unter Mehrere Netzwerkschnittstellen in AWS PCS.

Erstellen Sie eine Sicherheitsgruppe zur Unterstützung der EFA-Kommunikation

AWS CLI

Sie können den folgenden AWS CLI Befehl verwenden, um eine Sicherheitsgruppe zu erstellen, die EFA unterstützt. Der Befehl gibt eine Sicherheitsgruppen-ID aus. Nehmen Sie die folgenden Ersetzungen vor:

- *region-code*— Geben Sie an AWS-Region , wo Sie AWS PCS verwenden, z. B. `us-east-1`
- *vpc-id*— Geben Sie die ID der VPC an, die Sie für AWS PCS verwenden.
- *efa-group-name*— Geben Sie den von Ihnen gewählten Namen für die Sicherheitsgruppe ein.

```
aws ec2 create-security-group \
    --group-name efa-group-name \
    --description "Security group to enable EFA traffic" \
    --vpc-id vpc-id \
    --region region-code
```

Verwenden Sie die folgenden Befehle, um Sicherheitsgruppenregeln für eingehenden und ausgehenden Datenverkehr anzuhängen. Nehmen Sie den folgenden Ersatz vor:

- *efa-secgroup-id*— Geben Sie die ID der EFA-Sicherheitsgruppe an, die Sie gerade erstellt haben.

```
aws ec2 authorize-security-group-ingress \
    --group-id efa-secgroup-id \
    --protocol -1 \
    --source-group efa-secgroup-id

aws ec2 authorize-security-group-egress \
    --group-id efa-secgroup-id \
    --protocol -1 \
    --source-group efa-secgroup-id
```

CloudFormation template

Sie können eine CloudFormation Vorlage verwenden, um eine Sicherheitsgruppe zu erstellen, die EFA unterstützt. Laden Sie die Vorlage von der folgenden URL herunter und laden Sie sie dann in die AWS CloudFormation Konsole hoch.

```
https://aws-hpc-recipes.s3.amazonaws.com/main/recipes/pcs/enable_efa/assets/efa-sg.yaml
```

Geben Sie bei geöffneter Vorlage in der AWS CloudFormation Konsole die folgenden Optionen ein.

- Unter Geben Sie einen Stacknamen an
  - Geben Sie unter Stackname einen Namen ein, z. `efa-sg-stack` B.
- Unter Parameter
  - Geben Sie SecurityGroupNameunter einen Namen ein, z. `efa-sg` B.

- Wählen Sie unter VPC die VPC aus, in der Sie PCS verwenden AWS möchten.

Beenden Sie die Erstellung des CloudFormation Stacks und überwachen Sie seinen Status. Wenn es erreicht ist, ist `CREATE_COMPLETE` die EFA-Sicherheitsgruppe einsatzbereit.

(Optional) Erstellen Sie eine Platzierungsgruppe

Wir empfehlen, alle Instances, die EFA verwenden, in einer Cluster-Placement-Gruppe zu starten, um die physische Entfernung zwischen ihnen zu minimieren. Erstellen Sie eine Platzierungsgruppe für jede Rechenknotengruppe, in der Sie EFA verwenden möchten. Informationen Platzierungsgruppen für EC2-Instances in AWS PCS zum Erstellen einer Platzierungsgruppe für Ihre Compute-Knotengruppe finden Sie unter.

Erstellen oder aktualisieren Sie eine EC2-Startvorlage

EFA-Netzwerkschnittstellen werden in der EC2-Startvorlage für eine AWS PCS-Compute-Knotengruppe eingerichtet. Wenn mehrere Netzwerkkarten vorhanden sind, können mehrere EFAs konfiguriert werden. Die EFA-Sicherheitsgruppe und die optionale Platzierungsgruppe sind ebenfalls in der Startvorlage enthalten.

Hier ist ein Beispiel für eine Startvorlage für Instances mit zwei Netzwerkkarten, z. B. hpc7a.96xlarge. Die Instances werden in einer Cluster-Platzierungsgruppe gestartet. `subnet-`*`SubnetID1`* `pg-`*`PlacementGroupId1`*

Sicherheitsgruppen müssen jeder EFA-Schnittstelle speziell hinzugefügt werden. Jede EFA benötigt die Sicherheitsgruppe, die den EFA-Verkehr aktiviert (). `sg-`*`EfaSecGroupId`* Andere Sicherheitsgruppen, insbesondere solche, die regulären Datenverkehr wie SSH oder HTTPS verarbeiten, müssen nur an die primäre Netzwerkschnittstelle (gekennzeichnet durch ein `DeviceIndex` of) angehängt werden. `0` Startvorlagen, in denen Netzwerkschnittstellen definiert sind, unterstützen die Einstellung von Sicherheitsgruppen mithilfe des `SecurityGroupIds` Parameters nicht. Sie müssen `Groups` in jeder Netzwerkschnittstelle, die Sie konfigurieren, einen Wert für festlegen.

```
{
    "Placement": {
        "GroupId": "pg-PlacementGroupId1"
    },
    "NetworkInterfaces": [
```

```
        {
            "DeviceIndex": 0,
            "InterfaceType": "efa",
            "NetworkCardIndex": 0,
            "SubnetId": "subnet-SubnetId1",
            "Groups": [
                "sg-SecurityGroupId1",
                "sg-EfaSecGroupId"
            ]
        },
        {
            "DeviceIndex": 1,
            "InterfaceType": "efa",
            "NetworkCardIndex": 1,
            "SubnetId": "subnet-SubnetId1"
            "Groups": ["sg-EfaSecGroupId"]
        }
    ]
}
```

Erstellen oder aktualisieren Sie Compute-Knotengruppen für EFA

Ihre AWS PCS-Compute-Knotengruppen müssen Instances mit derselben Anzahl von vCPUs, Prozessorarchitektur und EFA-Unterstützung enthalten. Konfigurieren Sie die Compute-Knotengruppe so, dass sie das AMI mit der darauf installierten EFA-Software verwendet und die Startvorlage verwendet, mit der EFA-enabled Netzwerkschnittstellen konfiguriert werden.

(Optional) Testen Sie EFA

Sie können die EFA-enabled Kommunikation zwischen zwei Knoten in einer Rechenknotengruppe demonstrieren, indem Sie `fi_pingpong` das Programm ausführen, das in der EFA-Softwareinstallation enthalten ist. Wenn dieser Test erfolgreich ist, ist EFA wahrscheinlich richtig konfiguriert.

Zu Beginn benötigen Sie zwei laufende Instances in der Compute-Knotengruppe. Wenn Ihre Compute-Knotengruppe statische Kapazität verwendet, sollten bereits Instanzen verfügbar sein. Für eine Rechenknotengruppe, die dynamische Kapazität verwendet, können Sie mit dem `salloc` Befehl zwei Knoten starten. Hier ist ein Beispiel aus einem Cluster mit einer dynamischen Knotengruppe namens, die einer Warteschlange mit dem Namen `hpc7g` zugeordnet ist`all`.

```
% salloc --nodes 2 -p all
```

```
salloc: Granted job allocation 6
salloc: Waiting for resource configuration
... a few minutes pass ...
salloc: Nodes hpc7g-[1-2] are ready for job
```

Ermitteln Sie die IP-Adresse für die beiden zugewiesenen Knoten mithilfe von`scontrol`. Im folgenden Beispiel sind die Adressen `10.3.140.69` für `hpc7g-1` und `10.3.132.211` für`hpc7g-2`.

```
% scontrol show nodes hpc7g-[1-2]
NodeName=hpc7g-1 Arch=aarch64 CoresPerSocket=1
   CPUAlloc=0 CPUEfctv=64 CPUTot=64 CPULoad=0.00
   AvailableFeatures=hpc7g
   ActiveFeatures=hpc7g
   Gres=(null)
   NodeAddr=10.3.140.69 NodeHostName=ip-10-3-140-69 Version=25.11.2
   OS=Linux 6.12.80-106.156.amzn2023.aarch64 #1 SMP Fri May 1 14:08:14 UTC 2026
   RealMemory=124518 AllocMem=0 FreeMem=110763 Sockets=64 Boards=1
   State=IDLE+CLOUD ThreadsPerCore=1 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A
   Partitions=efa
   BootTime=2026-05-02T19:00:09 SlurmdStartTime=2026-05-08T19:33:25
   LastBusyTime=2026-05-08T19:33:25 ResumeAfterTime=None
   CfgTRES=cpu=64,mem=124518M,billing=64
   AllocTRES=
   CapWatts=n/a
   CurrentWatts=0 AveWatts=0
   ExtSensorsJoules=n/a ExtSensorsWatts=0 ExtSensorsTemp=n/a
   Reason=Maintain Minimum Number Of Instances [root@2026-05-02T18:59:00]
   InstanceId=i-04927897a9ce3c143 InstanceType=hpc7g.16xlarge

NodeName=hpc7g-2 Arch=aarch64 CoresPerSocket=1
   CPUAlloc=0 CPUEfctv=64 CPUTot=64 CPULoad=0.00
   AvailableFeatures=hpc7g
   ActiveFeatures=hpc7g
   Gres=(null)
   NodeAddr=10.3.132.211 NodeHostName=ip-10-3-132-211 Version=25.11.2
   OS=Linux 6.12.80-106.156.amzn2023.aarch64 #1 SMP Fri May 1 14:08:14 UTC 2026
   RealMemory=124518 AllocMem=0 FreeMem=110759 Sockets=64 Boards=1
   State=IDLE+CLOUD ThreadsPerCore=1 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A
   Partitions=efa
   BootTime=2026-05-02T19:00:09 SlurmdStartTime=2026-05-08T19:33:25
   LastBusyTime=2026-05-08T19:33:25 ResumeAfterTime=None
   CfgTRES=cpu=64,mem=124518M,billing=64
   AllocTRES=
```

```
   CapWatts=n/a
   CurrentWatts=0 AveWatts=0
   ExtSensorsJoules=n/a ExtSensorsWatts=0 ExtSensorsTemp=n/a
   Reason=Maintain Minimum Number Of Instances [root@2026-05-02T18:59:00]
   InstanceId=i-0a2c82623cb1393a7 InstanceType=hpc7g.16xlarge
```

Stellen Sie mithilfe von SSH (oder SSMhpc7g-1) eine Connect zu einem der Knoten her (in diesem Beispielfall). Beachten Sie, dass es sich um eine interne IP-Adresse handelt. Wenn Sie SSH verwenden, müssen Sie daher möglicherweise eine Verbindung von einem Ihrer Anmeldeknoten aus herstellen. Beachten Sie auch, dass die Instanz mithilfe der Startvorlage für Compute-Knotengruppen mit einem SSH-Schlüssel konfiguriert werden muss.

```
% ssh ec2-user@10.3.140.69
```

Starten Sie jetzt `fi_pingpong` im Servermodus.

```
/opt/amazon/efa/bin/fi_pingpong -p efa
```

Connect zur zweiten Instanz her (`hpc7g-2`).

```
% ssh ec2-user@10.3.132.211
```

Führen Sie `fi_pingpong` im Client-Modus aus und stellen Sie eine Verbindung zum Server her`hpc7g-1`. Sie sollten eine Ausgabe sehen, die dem Beispiel unten ähnelt.

```
% /opt/amazon/efa/bin/fi_pingpong -p efa 10.3.140.69

bytes   #sent   #ack     total       time     MB/sec    usec/xfer   Mxfers/sec
64      10      =10      1.2k        0.00s      3.08      20.75       0.05
256     10      =10      5k          0.00s     21.24      12.05       0.08
1k      10      =10      20k         0.00s     82.91      12.35       0.08
4k      10      =10      80k         0.00s    311.48      13.15       0.08
[error] util/pingpong.c:1876: fi_close (-22) fid 0
```

(Optional) Verwenden Sie eine CloudFormation Vorlage, um eine EFA-enabled Startvorlage zu erstellen

Da die Einrichtung von EFA mit mehreren Abhängigkeiten verbunden ist, wurde eine CloudFormation Vorlage bereitgestellt, mit der Sie eine Rechenknotengruppe konfigurieren können. Sie unterstützt

Instanzen mit bis zu vier Netzwerkkarten. Weitere Informationen zu Instances mit mehreren Netzwerkkarten finden Sie unter Elastic Network Interfaces im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Laden Sie die CloudFormation Vorlage von der folgenden URL herunter und laden Sie sie dann auf die CloudFormation Konsole hoch, AWS-Region in der Sie AWS PCS verwenden.

```
https://aws-hpc-recipes.s3.amazonaws.com/main/recipes/pcs/enable_efa/assets/pcs-lt-efa.yaml
```

Geben Sie bei geöffneter Vorlage in der CloudFormation Konsole die folgenden Werte ein. Beachten Sie, dass die Vorlage einige Standardparameterwerte bereitstellt. Sie können sie als Standardwerte beibehalten.

- Unter Geben Sie einen Stacknamen an
  - Geben Sie unter Stackname einen beschreibenden Namen ein. Wir empfehlen, den Namen zu verwenden, den Sie für Ihre AWS PCS-Compute-Knotengruppe wählen werden, z. B. *NODEGROUPNAME*`-efa-lt`
- Unter Parameter
  - Wählen Sie unter NumberOfNetworkCardsdie Anzahl der Netzwerkkarten in den Instanzen aus, die zu Ihrer Knotengruppe gehören sollen.
  - Wählen Sie unter die VPC aus VpcId, auf der Ihr AWS PCS-Cluster bereitgestellt wird.
  - Wählen Sie NodeGroupSubnetIdunter das Subnetz in Ihrer Cluster-VPC aus, in dem EFA-enabled Instances gestartet werden.
  - Lassen Sie das Feld unter leer PlacementGroupName, um eine neue Cluster-Platzierungsgruppe für die Knotengruppe zu erstellen. Wenn Sie bereits über eine Platzierungsgruppe verfügen, die Sie verwenden möchten, geben Sie hier ihren Namen ein.
  - Wählen Sie unter die Sicherheitsgruppe aus ClusterSecurityGroupId, die Sie verwenden, um den Zugriff auf andere Instances im Cluster und auf die AWS PCS-API zu gewähren. Viele Kunden wählen die Standardsicherheitsgruppe aus ihrer Cluster-VPC.
  - Geben Sie unter die ID für eine Sicherheitsgruppe ein SshSecurityGroupId, die Sie verwenden, um eingehenden SSH-Zugriff auf Knoten in Ihrem Cluster zu ermöglichen.
  - Wählen Sie für SshKeyNamedas SSH-Schlüsselpaar für den Zugriff auf Knoten in Ihrem Cluster aus.

  - Geben Sie für LaunchTemplateNameeinen beschreibenden Namen für die Startvorlage ein, z. B. *NODEGROUPNAME*`-efa-lt` Der Name muss für den Ort, AWS-Konto an AWS-Region dem Sie AWS PCS verwenden werden, einzigartig sein.
- Unter Funktionen
  - Markieren Sie das Kästchen Ich bestätige, dass dadurch IAM-Ressourcen erstellt werden AWS CloudFormation könnten.

Überwachen Sie den Status des CloudFormation Stacks. Wenn `CREATE_COMPLETE` die Startvorlage erreicht ist, kann sie verwendet werden. Verwenden Sie es mit einer AWS PCS-Compute-Knotengruppe, wie oben unter beschriebenErstellen oder aktualisieren Sie Compute-Knotengruppen für EFA.

Verwenden von Netzwerkdateisystemen mit AWS PCS

Sie können Netzwerkdateisysteme an Knoten anhängen, die in einer AWS PCS-Rechenknotengruppe (AWS Parallel Computing Service) gestartet wurden, um einen dauerhaften Speicherort bereitzustellen, an dem Daten und Dateien geschrieben und abgerufen werden können. Sie können Dateisysteme verwenden, die von AWS Diensten wie Amazon Elastic File System (Amazon EFS), Amazon FSx for Lustre, Amazon for NetApp ONTAP, Amazon FSx FSx forOpenZFS und Amazon File Cache bereitgestellt werden. Sie können auch selbstverwaltete Dateisysteme wie NFS-Server verwenden.

In diesem Thema werden Überlegungen und Beispiele für die Verwendung von Netzwerkdateisystemen mit AWS PCS behandelt.

Überlegungen zur Verwendung von Netzwerkdateisystemen

Die Implementierungsdetails für verschiedene Dateisysteme sind unterschiedlich, es gibt jedoch einige allgemeine Überlegungen.

- Die entsprechende Dateisystemsoftware muss auf der Instanz installiert sein. Um beispielsweise Amazon FSx for Lustre zu verwenden, sollte das entsprechende Lustre Paket vorhanden sein. Dies kann erreicht werden, indem es in das Compute-Knotengruppen-AMI aufgenommen wird oder indem ein Skript verwendet wird, das beim Instance-Start ausgeführt wird.
- Es muss eine Netzwerkroute zwischen dem gemeinsam genutzten Netzwerkdateisystem und den Compute-Knotengruppen-Instances bestehen.
- Die Sicherheitsgruppenregeln sowohl für das gemeinsam genutzte Netzwerk-Dateisystem als auch für die Compute-Knotengruppen-Instanzen müssen Verbindungen zu den entsprechenden Ports zulassen.
- Sie müssen für alle Ressourcen, die auf die Dateisysteme zugreifen, einen konsistenten POSIX Benutzer- und Gruppennamespace aufrechterhalten. Andernfalls kann es bei Aufträgen und interaktiven Prozessen, die auf Ihrem PCS-Cluster ausgeführt werden, zu Berechtigungsfehlern kommen.
- Das Einhängen von Dateisystemen erfolgt mithilfe von EC2 Startvorlagen. Fehler oder Zeitüberschreitungen beim Mounten eines Netzwerkdateisystems können dazu führen, dass Instanzen nicht mehr für die Ausführung von Jobs verfügbar sind. Dies wiederum kann zu unerwarteten Kosten führen. Weitere Informationen zum Debuggen von Startvorlagen finden Sie unterVerwenden von Amazon EC2 EC2-Startvorlagen mit AWS PCS.

Beispiele für Netzwerk-Mounts

Sie können Dateisysteme mit Amazon EFS, Amazon FSx for Lustre, Amazon for NetApp ONTAP, Amazon FSx FSx for OpenZFS und Amazon File Cache erstellen. Erweitern Sie den entsprechenden Abschnitt unten, um ein Beispiel für jeden Netzwerk-Mount zu sehen.

Amazon EFS

Einrichtung des Dateisystems

Erstellen Sie ein Amazon EFS-Dateisystem. Stellen Sie sicher, dass es in jeder Availability Zone, in der Sie PCS-Compute-Knotengruppen-Instances starten, ein Mount-Ziel gibt. Stellen Sie außerdem sicher, dass jedes Mount-Ziel einer Sicherheitsgruppe zugeordnet ist, die eingehenden und ausgehenden Zugriff von den PCS-Compute-Knotengruppen-Instances aus ermöglicht. Weitere Informationen finden Sie unter Bereitstellen von Zielen und Sicherheitsgruppen im Amazon Elastic File System-Benutzerhandbuch.

Startvorlage

Fügen Sie die Sicherheitsgruppe (n) aus Ihrem Dateisystem-Setup zur Startvorlage hinzu, die Sie für die Compute-Knotengruppe verwenden werden.

Fügen Sie Benutzerdaten hinzu, die `cloud-config` einen Mechanismus zum Mounten des Amazon EFS-Dateisystems verwenden. Ersetzen Sie die folgenden Werte in diesem Skript durch Ihre eigenen Daten:

- *`mount-point-directory`*— Der Pfad auf jeder Instance, auf der Sie Amazon EFS mounten werden
- *`filesystem-id`*— Die Dateisystem-ID für das EFS-Dateisystem

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="==MYBOUNDARY=="

--==MYBOUNDARY==
Content-Type: text/cloud-config; charset="us-ascii"

packages:
  - amazon-efs-utils
```

```
runcmd:
  - mkdir -p /mount-point-directory
  - echo "filesystem-id:/ /mount-point-directory efs tls,_netdev" >> /etc/fstab
  - mount -a -t efs defaults

--==MYBOUNDARY==--
```

Amazon FSx für Lustre

Einrichtung des Dateisystems

Erstellen Sie ein FSx for Lustre-Dateisystem in der VPC, in dem Sie PCS verwenden AWS werden. Um Übertragungen zwischen Zonen zu minimieren, sollten Sie die Implementierung in einem Subnetz in derselben Availability Zone durchführen, in der Sie die meisten Ihrer PCS-Compute-Knotengruppen-Instances starten werden. Stellen Sie sicher, dass das Dateisystem einer Sicherheitsgruppe zugeordnet ist, die eingehenden und ausgehenden Zugriff von den PCS-Compute-Knotengruppen-Instances aus ermöglicht. Weitere Informationen zu Sicherheitsgruppen finden Sie unter Dateisystem-Zugriffskontrolle mit Amazon VPC im Amazon FSx for Lustre-Benutzerhandbuch.

Startvorlage

Fügen Sie Benutzerdaten hinzu, die `cloud-config` zum Mounten des FSx for Lustre-Dateisystems verwendet werden. Ersetzen Sie die folgenden Werte in diesem Skript durch Ihre eigenen Daten:

- *`mount-point-directory`*— Der Pfad auf einer Instanz, die Sie FSx für Lustre mounten möchten
- *`filesystem-id`*— Die Dateisystem-ID für das FSx for Lustre-Dateisystem
- *`mount-name`*— Der Mount-Name für das FSx for Lustre-Dateisystem
- *`region-code`*— Der AWS-Region Ort, an dem das FSx for Lustre-Dateisystem bereitgestellt wird (muss mit Ihrem AWS PCS-System identisch sein)
- (Optional) *`latest`* — Jede Version von, die von FSx for Lustre Lustre unterstützt wird

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="==MYBOUNDARY=="

--==MYBOUNDARY==
Content-Type: text/cloud-config; charset="us-ascii"

runcmd:
- amazon-linux-extras install -y lustre=latest
```

```
- mkdir -p /mount-point-directory
- mount -t lustre filesystem-id.fsx.region-code.amazonaws.com@tcp:/mount-name /mount-
point-directory

--==MYBOUNDARY==--
```

Amazon FSx für NetApp ONTAP

Einrichtung des Dateisystems

Erstellen Sie ein Amazon FSx for NetApp ONTAP-Dateisystem in der VPC, in der Sie PCS verwenden AWS werden. Um Übertragungen zwischen Zonen zu minimieren, sollten Sie die Bereitstellung in einem Subnetz in derselben Availability Zone durchführen, in der Sie die meisten Ihrer AWS PCS-Compute-Knotengruppen-Instances starten werden. Stellen Sie sicher, dass das Dateisystem einer Sicherheitsgruppe zugeordnet ist, die eingehenden und ausgehenden Zugriff von den Instances der AWS PCS-Compute-Knotengruppe aus ermöglicht. Weitere Informationen zu Sicherheitsgruppen finden Sie unter File System Access Control with Amazon VPC im FSx for ONTAP User Guide.

Startvorlage

Fügen Sie Benutzerdaten hinzu, die `cloud-config` zum Mounten des Root-Volumes FSx für ein ONTAP-Dateisystem verwendet werden. Ersetzen Sie die folgenden Werte in diesem Skript durch Ihre eigenen Daten:

- *`mount-point-directory`*— Der Pfad auf einer Instance, auf der Sie Ihr FSx for ONTAP-Volume mounten möchten
- *`svm-id`*— Die SVM-ID für das Dateisystem FSx für ONTAP
- *`filesystem-id`*— Die Dateisystem-ID FSx für das ONTAP-Dateisystem
- *`region-code`*— Der AWS-Region Ort, an dem das Dateisystem FSx für ONTAP bereitgestellt wird (muss mit Ihrem AWS PCS-System identisch sein)
- *`volume-name`*— Der Name des FSx Volumes für ONTAP

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="==MYBOUNDARY=="

--==MYBOUNDARY==
Content-Type: text/cloud-config; charset="us-ascii"
```

```
runcmd:
- mkdir -p /mount-point-directory
- mount -t nfs svm-id.filesystem-id.fsx.region-code.amazonaws.com:/volume-name /mount-
point-directory

--==MYBOUNDARY==--
```

Amazon FSx für OpenZFS

Einrichtung des Dateisystems

Erstellen Sie ein Dateisystem FSx für OpenZFS in der VPC, in dem Sie PCS verwenden werden. AWS Um Übertragungen zwischen Zonen zu minimieren, sollten Sie die Implementierung in einem Subnetz in derselben Availability Zone durchführen, in der Sie die meisten Ihrer AWS PCS-Compute-Knotengruppen-Instances starten werden. Stellen Sie sicher, dass das Dateisystem einer Sicherheitsgruppe zugeordnet ist, die eingehenden und ausgehenden Zugriff von den Instances der AWS PCS-Compute-Knotengruppe aus ermöglicht. Weitere Informationen zu Sicherheitsgruppen finden Sie unter Verwaltung des Dateisystemzugriffs mit Amazon VPC im FSx OpenZFS-Benutzerhandbuch.

Startvorlage

Fügen Sie Benutzerdaten hinzu, die `cloud-config` zum Mounten des Root-Volumes FSx für ein OpenZFS-Dateisystem verwendet werden. Ersetzen Sie die folgenden Werte in diesem Skript durch Ihre eigenen Daten:

- *`mount-point-directory`*— Der Pfad auf einer Instanz, auf der Sie Ihr FSx für OpenZFS Share mounten möchten
- *`filesystem-id`*— Die Dateisystem-ID für das Dateisystem FSx für OpenZFS
- *`region-code`*— Der AWS-Region Ort, an dem das Dateisystem FSx für OpenZFS bereitgestellt wird (muss mit Ihrem PCS-System identisch sein) AWS

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="==MYBOUNDARY=="

--==MYBOUNDARY==
Content-Type: text/cloud-config; charset="us-ascii"
```

```
runcmd:
- mkdir -p /mount-point-directory
- mount -t nfs -o noatime,nfsvers=4.2,sync,rsize=1048576,wsize=1048576 filesystem-
id.fsx.region-code.amazonaws.com:/fsx/ /mount-point-directory

--==MYBOUNDARY==--
```

Amazon-Datei-Cache

Einrichtung des Dateisystems

Erstellen Sie einen Amazon File Cache in der VPC, in der Sie AWS PCS verwenden werden. Um Übertragungen zwischen Zonen zu minimieren, wählen Sie ein Subnetz in derselben Availability Zone aus, in der Sie die meisten Ihrer PCS-Compute-Knotengruppen-Instances starten werden. Stellen Sie sicher, dass der Datei-Cache einer Sicherheitsgruppe zugeordnet ist, die eingehenden und ausgehenden Datenverkehr auf Port 988 zwischen Ihren PCS-Instances und dem File Cache zulässt. Weitere Informationen zu Sicherheitsgruppen finden Sie unter Cache-Zugriffskontrolle mit Amazon VPC im Amazon File Cache-Benutzerhandbuch.

Startvorlage

Fügen Sie die Sicherheitsgruppe (n) aus Ihrem Dateisystem-Setup zur Startvorlage hinzu, die Sie für die Compute-Knotengruppe verwenden werden.

Schließen Sie Benutzerdaten ein, die `cloud-config` zum Mounten des Amazon File Cache verwendet werden. Ersetzen Sie die folgenden Werte in diesem Skript durch Ihre eigenen Daten:

- *`mount-point-directory`*— Der Pfad auf einer Instanz, die Sie FSx für Lustre mounten möchten
- *`cache-dns-name`*— Der DNS-Name (Domain Name System) für den Dateicache
- *`mount-name`*— Der Mount-Name für den Datei-Cache

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="==MYBOUNDARY=="

--==MYBOUNDARY==
Content-Type: text/cloud-config; charset="us-ascii"

runcmd:
- amazon-linux-extras install -y lustre=2.12
- mkdir -p /mount-point-directory
```

```
- mount -t lustre -o relatime,flock cache-dns-name@tcp:/mount-name /mount-point-
directory

--==MYBOUNDARY==--
```

Amazon Machine Images (AMIs) für AWS STK.

AWS PCS arbeitet mit AMIs, die Sie bereitstellen, und bietet so eine große Flexibilität bei der Software und Konfiguration der Knoten in Ihrem Cluster. Für Produktions AI/ML - und HPC-Workloads können Sie das AWS-maintained PCS-ready DLAMI verwenden. Wenn Sie AWS PCS evaluieren, können Sie ein Beispiel-AMI verwenden, das von bereitgestellt wird AWS. Sie können auch Ihre eigenen benutzerdefinierten AMIs erstellen, um die volle Kontrolle über die Software und Konfiguration auf Ihren Knoten zu haben.

Themen

Verwenden von PCS-ready DLAMI mit AWS STK.

AWS PCS-ready DLAMI Base GPU AMI (Ubuntu 24.04) ist ein AWS-maintained Amazon Machine Image für die Ausführung AI/ML und HPC-Workloads auf PCs. AWS Es bietet eine produktionsbereite Grundlage, sodass Sie Cluster innerhalb von Minuten bereitstellen können, anstatt benutzerdefinierte AMIs zu erstellen und zu validieren.

Was ist enthalten

PCS-ready DLAMI basiert auf dem Deep Learning Base GPU AMI (Ubuntu 24.04) und fügt die folgenden PCS-Komponenten hinzu: AWS

- PCS Agent — Der AWS PCS-Clusterverwaltungsagent
- Slurm for AWS PCS — Mehrere unterstützte Slurm-Versionen sind vorinstalliert. Die richtige Version wird beim Start der Instanz automatisch aktiviert, basierend auf der Konfiguration Ihres Clusters.
- EFS-Dienstprogramme — Zum Mounten von Amazon EFS-Dateisystemen

Das Quell-DLAMI stellt das Betriebssystem (Ubuntu 24.04), NVIDIA-GPU-Treiber, das CUDA-Toolkit, die EFA-Treiber, den Lustre-Client und andere grundlegende Infrastrukturen bereit. Einzelheiten zu diesen Komponenten finden Sie in den Deep Learning AMI-Versionshinweisen.

PCS-ready DLAMI ist sowohl für x86_64- als auch für arm64-Architekturen verfügbar.

> **Note**
>
> PCS-ready DLAMI umfasst keine Anwendungssoftware wie AI/ML Frameworks (PyTorch,, JAX) TensorFlow, Compiler oder mathematische Bibliotheken. Sie können Ihre Anwendungsebene auf gemeinsam genutzten Dateisystemen hinzufügen oder indem Sie ein benutzerdefiniertes AMI auf PCS-ready DLAMI aufbauen.

Das Beschreibungsfeld jedes AMI fasst seinen Inhalt zusammen, einschließlich der Quell-DLAMI, auf der es basiert, der PCS-Agent-Version, der unterstützten Slurm-Versionen und der Version der EFS-Dienstprogramme. Sie können dieses Feld in der Amazon EC2 EC2-Konsole oder mithilfe der `describe-images` API anzeigen. Im Folgenden finden Sie ein Beispiel für einen Wert im Feld Beschreibung:

```
PCS-Ready DLAMI based on Deep Learning Base OSS Nvidia Driver GPU AMI (Ubuntu 24.04)
 20260522. PCS Agent: 1.4.0-1. Slurm: 24.11.7-1, 25.05.7-1, 25.11.2-1. EFS Utils: 2.4.2
```

Finden Sie das aktuelle PCS-ready DLAMI

AWS-Managementkonsole

Um PCS-ready DLAMI in der Konsole zu finden

1. Öffnen Sie die AWS PCS-Konsole und navigieren Sie zum Erstellen oder Bearbeiten einer Rechenknotengruppe.
2. Wählen Sie im Abschnitt AMI-Auswahl die Option PCS-ready AMIs aus.
3. Es erscheint eine Dropdownliste mit verfügbaren PCS-ready DLAMIs, gefiltert nach der Architektur Ihres ausgewählten Instance-Typs.
4. Wählen Sie AWS PCS-ready DLAMI Base AMI (Ubuntu 24.04). In der Dropdownliste werden die AMI-ID und der vollständige AMI-Name unten als Referenz angezeigt.

AWS CLI

Sie können die neueste PCS-ready DLAMI-AMI-ID mithilfe des Amazon EC2 Systems Manager Parameter Store abrufen. Ersetzen Sie durch Ihre*region-code*. AWS-Region

- x86_64

```
aws ssm get-parameter --region region-code \
  --name /aws/service/pcs/ami/dlami-base-ubuntu2404/x86_64/latest/ami-id \
  --query "Parameter.Value" --output text
```

- arm64

```
aws ssm get-parameter --region region-code \
  --name /aws/service/pcs/ami/dlami-base-ubuntu2404/arm64/latest/ami-id \
  --query "Parameter.Value" --output text
```

Alternativ können Sie anhand des Namensmusters nach PCS-ready DLAMI suchen:

- x86_64

```
aws ec2 describe-images --region region-code --owners amazon \
  --filters 'Name=name,Values=aws-pcs-ready-dlami-base-ubuntu2404-x86_64-*' \
            'Name=state,Values=available' \
  --query 'sort_by(Images, &CreationDate)[-1].[Name,ImageId]' --output text
```

- arm64

```
aws ec2 describe-images --region region-code --owners amazon \
  --filters 'Name=name,Values=aws-pcs-ready-dlami-base-ubuntu2404-arm64-*' \
            'Name=state,Values=available' \
  --query 'sort_by(Images, &CreationDate)[-1].[Name,ImageId]' --output text
```

Verwenden Sie die AMI-ID, wenn Sie eine Compute-Knotengruppe erstellen oder aktualisieren.

Zusammen mit Infrastructure as Code verwenden

Der SSM-Parameterpfad bietet eine stabile Referenz, die immer zur neuesten AMI-ID aufgelöst wird. Sie können dies in CloudFormation Vorlagen verwenden, um neue Versionen bei der erneuten Bereitstellung automatisch zu übernehmen:

```
AmiId: '{{resolve:ssm:/aws/service/pcs/ami/dlami-base-ubuntu2404/x86_64/latest/ami-id}}'
```

Auf eine neue Version aktualisieren

AWS veröffentlicht aktualisierte PCS-ready DLAMI-Versionen, wenn das Deep Learning Base GPU AMI der Quelle aktualisiert wird oder wenn PCS-Komponenten (PCS Agent oder Slurm for PCS) aktualisiert werden. Um Ihren Cluster zu aktualisieren, rufen Sie die neueste AMI-ID mithilfe des oben beschriebenen SSM-Parameters oder der Namenssuche ab und aktualisieren Sie dann jede Rechenknotengruppe, sodass sie auf die neue AMI-ID verweist.

Verwenden von Amazon Machine Images (AMIs) -Beispieldateien mit AWS STK.

AWS stellt Beispiel-AMIs bereit, die Sie als Ausgangspunkt für die Arbeit mit AWS PCS verwenden können.

> **Important**
>
> Beispiel-AMIs dienen zu Demonstrationszwecken und werden nicht für Produktionsworkloads empfohlen.

> **Important**
>
> Compute-Knotengruppen, die mit AWS PCS-Beispiel-AMIs und mehreren Netzwerkschnittstellen konfiguriert sind, funktionieren derzeit nicht, wenn die Subnetze nur für die Verwendung von IPv6 konfiguriert sind. Verwenden Sie stattdessen Dual-Stack-Subnetze (IPv4 und IPv6) oder Subnetze. IPv4-only

Finden Sie aktuelle AWS PCS-Beispiel-AMIs

AWS-Managementkonsole

AWS-PCS-Beispiel-AMIs haben die folgende Benennungskonvention:

```
aws-pcs-sample_ami-OS-architecture-scheduler-scheduler-major-version
```

Akzeptierte Werte

- *OS* – `al2023`
- *architecture* – `x86_64` oder `arm64`
- *scheduler* – `slurm`
- *scheduler-major-version* – `25.11`

Um zu finden AWS PCS-Beispiel-AMIs

1. Öffnen Sie die Amazon EC2-Konsole.
2. Navigieren Sie zu AMIs.
3. Wählen Sie Öffentliche Abbilder aus.
4. Suchen Sie unter AMI nach Attribut oder Tag suchen Sie anhand des Vorlagennamens nach einem AMI.

   Beispiele

   - Beispiel-AMI für Slurm 25.11 auf Arm64-Instances

     ```
     aws-pcs-sample_ami-al2023-arm64-slurm-25.11
     ```

   - Beispiel-AMI für Slurm 25.11 auf x86-Instances

     ```
     aws-pcs-sample_ami-al2023-x86_64-slurm-25.11
     ```

   > **Note**
   >
   > Wenn es mehrere AMIs gibt, verwenden Sie das AMI mit dem neuesten Zeitstempel.

> **Note**
>
> Beispiel-AMIs für Slurm 25.05 verwenden Amazon Linux 2 (`amzn2`) anstelle von Amazon Linux 2023 (). `al2023`

5. Verwenden Sie die AMI-ID, wenn Sie eine Compute-Knotengruppe erstellen oder aktualisieren.

AWS CLI

Sie finden das neueste AWS PCS-Beispiel-AMI mit den folgenden Befehlen. *`region-code`*Ersetzen Sie es durch das, AWS-Region wo Sie AWS PCS verwenden, z. `us-east-1` B.

- x86_64

```
aws ec2 describe-images --region region-code --owners amazon \
--filters 'Name=name,Values=aws-pcs-sample_ami-al2023-x86_64-slurm-25.11*' \
            'Name=state,Values=available' \
--query 'sort_by(Images, &CreationDate)[-1].[Name,ImageId]' --output text
```

- Arm 64

```
aws ec2 describe-images --region region-code --owners amazon \
--filters 'Name=name,Values=aws-pcs-sample_ami-al2023-arm64-slurm-25.11*' \
            'Name=state,Values=available' \
--query 'sort_by(Images, &CreationDate)[-1].[Name,ImageId]' --output text
```

Verwenden Sie die AMI-ID, wenn Sie eine Compute-Knotengruppe erstellen oder aktualisieren.

Erfahren Sie mehr über AWS PCS-Beispiel-AMIs

Den Inhalt und die Konfigurationsdetails für aktuelle und frühere Versionen der AWS PCS-Beispiel-AMIs finden Sie unterVersionshinweise für AWS PCS-Beispiel-AMIs.

Erstellen Sie Ihre eigenen AMIs, die kompatibel sind mit AWS STK.

Informationen zum Erstellen eigener AMIs, die mit AWS PCS funktionieren, finden Sie unterBenutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS.

Benutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS

AWS PCS ist so konzipiert, dass es mit Amazon Machine Images (AMI) funktioniert, die Sie für den Service bereitstellen. Auf diesen AMIs können beliebige Software und Konfigurationen installiert sein, sofern auf ihnen der AWS PCS-Agent und eine kompatible Version von Slurm korrekt installiert und konfiguriert sind. Sie müssen die von Ihnen AWS bereitgestellten Installationsprogramme verwenden, um die AWS PCS-Software auf Ihrem benutzerdefinierten AMI zu installieren. Wir empfehlen Ihnen, AWS zur Installation von Slurm auf Ihrem benutzerdefinierten AMI bereitgestellte Installationsprogramme zu verwenden, aber Sie können Slurm auch selbst installieren, wenn Sie dies bevorzugen (nicht empfohlen).

> **Note**
>
> Wenn Sie AWS PCS ausprobieren möchten, ohne ein benutzerdefiniertes AMI zu erstellen, können Sie ein Beispiel-AMI verwenden, das von bereitgestellt wird AWS. Weitere Informationen finden Sie unter Verwenden von Amazon Machine Images (AMIs) - Beispieldateien mit AWS STK..

> **Important**
>
> AWS PCS benötigt derzeit einen Kernel mit IPv4 Unterstützung für lokale Knotenkommunikation, auch wenn Sie AWS PCS in einem IPv6 reinen Netzwerk verwenden.

Dieses Tutorial hilft Ihnen bei der Erstellung eines AMI, das mit PCS-Rechenknotengruppen verwendet werden kann, um Ihr HPC und Ihre AI/ML Workloads zu unterstützen.

Themen

Schritt 1 — Eine temporäre Instanz starten

Starten Sie eine temporäre Instanz, mit der Sie die AWS PCS-Software und den Slurm-Scheduler installieren und konfigurieren können. Sie verwenden diese Instance, um ein mit AWS PCS kompatibles AMI zu erstellen.

So starten Sie eine temporäre Instance

1. Öffnen Sie die Amazon EC2-Konsole.
2. Wählen Sie im Navigationsbereich Instances und anschließend Launch Instances aus, um den Assistenten zum Starten neuer Instances zu öffnen.
3. (Optional) Geben Sie im Abschnitt Name und Tags einen Namen für die Instance ein, z. `PCS-AMI-instance` B. Der Name wird der Instance als Ressourcen-Tag (`Name=PCS-AMI-instance`) zugewiesen.
4. Wählen Sie im Bereich Application and OS Images (Anwendungs- und Betriebssystem-Images) ein AMI für eines der unterstützten Betriebssysteme aus.
5. Wählen Sie im Bereich Instance type (Instance-Typ) einen supported instance type (unterstützten Instance-Typ) aus.
6. Wählen Sie im Bereich Key pair (Schlüsselpaar) das Schlüsselpaar aus, das für die Instance verwendet werden soll.
7. Gehen Sie im Abschnitt Netzwerkeinstellungen wie folgt vor:
    - Wählen Sie für Firewall (Sicherheitsgruppen) die Option Bestehende Sicherheitsgruppe auswählen und anschließend eine Sicherheitsgruppe aus, die eingehenden SSH-Zugriff auf Ihre Instance ermöglicht.
8. Konfigurieren Sie im Bereich Storage (Speicher) die Volumes nach Bedarf. Stellen Sie sicher, dass ausreichend Speicherplatz für die Installation Ihrer eigenen Anwendungen und Bibliotheken konfiguriert ist.
9. Wählen Sie in der Übersicht Launch instance (Instance starten) aus.

Schritt 2 — Installieren Sie den AWS PCS-Agenten

Installieren Sie den Agenten, der die von AWS PCS gestarteten Instanzen für die Verwendung mit Slurm konfiguriert. Weitere Informationen zum AWS PCS-Agenten finden Sie unter. AWS Versionen von PCS-Agenten

Um den AWS PCS-Agenten zu installieren

1. Stellen Sie eine Verbindung zu der Instance her, die Sie gestartet haben. Weitere Informationen finden Sie unter Connect zu Ihrer Linux-Instance herstellen.
2. (Optional) Um sicherzustellen, dass alle Ihre Softwarepakete auf dem neuesten Stand sind, führen Sie ein schnelles Softwareupdate auf Ihrer Instance durch. Dieser Vorgang kann einige Minuten dauern.

   - Amazon Linux 2, Amazon Linux 2023, RHEL 9, RHEL 8, Rocky Linux 9 und Rocky Linux 8

     ```
     sudo yum update -y
     ```

   - Ubuntu 22.04 und Ubuntu 24.04

     ```
     sudo apt-get update && sudo apt-get upgrade -y
     ```

3. Starten Sie die Instance neu und stellen Sie die Verbindung zur Instance wieder her.
4. Laden Sie die Installationsdateien für den AWS PCS-Agenten herunter. Die Installationsdateien sind in eine komprimierte Tarball-Datei (`.tar.gz`) gepackt. Laden Sie die neueste stabile Version mit dem folgenden Befehl herunter. *`region`*Ersetzen Sie es durch den AWS-Region Ort, an dem Sie Ihre temporäre Instance gestartet haben, z. B. `us-east-1`

   ```
   curl https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.4.0-1.tar.gz -o aws-pcs-agent-v1.4.0-1.tar.gz
   ```

   Sie können die neueste Version auch abrufen, indem Sie die Versionsnummer durch `latest` den vorherigen Befehl ersetzen (zum Beispiel:`aws-pcs-agent-v1-latest.tar.gz`).

   > **Note**
   >
   > Dies könnte sich in future Versionen der AWS PCS-Agent-Software ändern.

5. (Optional) Überprüfen Sie die Authentizität und Integrität des AWS PCS-Software-Tarballs. Diese Vorgehensweise wird empfohlen, um die Identität des Software-Publishers zu überprüfen und sicherzustellen, dass die Datei seit ihrer Veröffentlichung nicht verändert oder beschädigt wurde.

   a. Laden Sie den öffentlichen GPG-Schlüssel für AWS PCS herunter und importieren Sie ihn in Ihren Schlüsselbund. Ersetzen Sie ihn *region* durch den AWS-Region Ort, an dem Sie Ihre temporäre Instance gestartet haben. Der Befehl sollte einen Schlüsselwert zurückgeben. Notieren Sie sich den Schlüsselwert. Sie verwenden ihn im nächsten Schritt.

   ```
   wget https://aws-pcs-repo-public-keys-region.s3.region.amazonaws.com/aws-pcs-public-key.pub && \
       gpg --import aws-pcs-public-key.pub
   ```

   b. Führen Sie den folgenden Befehl aus, um den Fingerabdruck des GPG-Schlüssels zu überprüfen.

   ```
   gpg --fingerprint 7EEF030EDDF5C21C
   ```

   Der Befehl sollte einen Fingerabdruck zurückgeben, der mit dem folgenden identisch ist:

   ```
   1C24 32C1 862F 64D1 F90A  239A 7EEF 030E DDF5 C21C
   ```

   > **Important**
   >
   > Führen Sie das Installationsskript für den AWS PCS-Agenten nicht aus, wenn der Fingerabdruck nicht übereinstimmt. AWS Support kontaktieren.

   c. Laden Sie die Signaturdatei herunter und überprüfen Sie die Signatur der AWS PCS-Software-Tarball-Datei. *region*Ersetzen Sie durch den AWS-Region Ort, an dem Sie Ihre temporäre Instance gestartet haben, z. B. `us-east-1`

   ```
   wget https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.4.0-1.tar.gz.sig && \
       gpg --verify ./aws-pcs-agent-v1.4.0-1.tar.gz.sig
   ```

   Die Ausgabe sollte folgendermaßen oder ähnlich aussehen:

   ```
   gpg: assuming signed data in './aws-pcs-agent-v1.4.0-1.tar.gz'
   ```

```
gpg: Signature made Thu 06 Nov 2025 11:10:36 AM CET using RSA key ID ECC0AE5C
gpg: Good signature from "AWS PCS Packages (AWS PCS Packages)"
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 1C24 32C1 862F 64D1 F90A  239A 7EEF 030E DDF5 C21C
   Subkey fingerprint: B7E1 8788 3517 6A74 C3D5  EAF5 6088 136D ECC0 AE5C
```

Wenn das Ergebnis den Fingerabdruck enthält `Good signature` und der Fingerabdruck mit dem im vorherigen Schritt zurückgegebenen Fingerabdruck übereinstimmt, fahren Sie mit dem nächsten Schritt fort.

> **Important**
>
> Führen Sie das AWS PCS-Softwareinstallationsskript nicht aus, wenn der Fingerabdruck nicht übereinstimmt. AWS Support kontaktieren.

6. Extrahieren Sie die Dateien aus der komprimierten `.tar.gz` Datei und navigieren Sie zum entpackten Verzeichnis.

```
tar -xf aws-pcs-agent-v1.4.0-1.tar.gz && \
    cd aws-pcs-agent
```

7. Installieren Sie die AWS PCS-Software.

```
sudo ./installer.sh
```

8. Überprüfen Sie die Versionsdatei der AWS PCS-Software, um zu bestätigen, dass die Installation erfolgreich war.

```
cat /opt/aws/pcs/version
```

Die Ausgabe sollte folgendermaßen oder ähnlich aussehen:

```
AGENT_INSTALL_DATE='Fri Dec 13 12:28:43 UTC 2024'
AGENT_VERSION='1.4.0'
AGENT_RELEASE='1'
```

Schritt 3 — Slurm installieren

Installieren Sie eine Version von Slurm, die mit AWS PCS kompatibel ist. Weitere Informationen finden Sie unter Slurm-Versionen in AWS STK..

> **Note**
>
> Wenn Sie ein AMI haben, auf dem eine frühere Version der Slurm-Software installiert ist, müssen Sie die folgenden Schritte ausführen, um die neue Version von Slurm zu installieren. Der AWS PCS-Agent aktiviert zur Laufzeit die richtige Version der Slurm-Binärdateien entsprechend der Slurm-Version, die zum Zeitpunkt der Clustererstellung konfiguriert wurde.

Um Slurm zu installieren

1. Connect zu derselben temporären Instanz her, auf der Sie die AWS PCS-Software installiert haben.
2. Laden Sie die Slurm-Installationssoftware herunter. Der Slurm-Installer ist in eine komprimierte Tarball () `.tar.gz` -Datei gepackt. Laden Sie die neueste stabile Version mit dem folgenden Befehl herunter. *`region`*Ersetzen Sie es durch die AWS-Region Ihrer temporären Instanz, z. B. `us-east-1`

   ```
   curl https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-25.11-installer-25.11.2-1.tar.gz \
        -o aws-pcs-slurm-25.11-installer-25.11.2-1.tar.gz
   ```

   Sie können die neueste Version auch abrufen, indem Sie die Versionsnummer durch `latest` den vorherigen Befehl ersetzen (zum Beispiel:`aws-pcs-slurm-25.11-installer-latest.tar.gz`). Eine vollständige Liste der verfügbaren Versionen mit Prüfsummen finden Sie unterSlurm-Versionen in AWS STK..

   > **Note**
   >
   > Dies könnte sich in future Versionen der Slurm-Installationssoftware ändern.

3. (Optional) Überprüfen Sie die Authentizität und Integrität des Slurm-Installations-Tarballs. Diese Vorgehensweise wird empfohlen, um die Identität des Software-Publishers zu überprüfen und sicherzustellen, dass die Datei seit ihrer Veröffentlichung nicht verändert oder beschädigt wurde.

a. Laden Sie den öffentlichen GPG-Schlüssel für AWS PCS herunter und importieren Sie ihn in Ihren Schlüsselbund. Ersetzen Sie ihn *region* durch den AWS-Region Ort, an dem Sie Ihre temporäre Instance gestartet haben. Der Befehl sollte einen Schlüsselwert zurückgeben. Notieren Sie sich den Schlüsselwert. Sie verwenden ihn im nächsten Schritt.

```
wget https://aws-pcs-repo-public-keys-region.s3.region.amazonaws.com/aws-pcs-public-key.pub && \
    gpg --import aws-pcs-public-key.pub
```

b. Führen Sie den folgenden Befehl aus, um den Fingerabdruck des GPG-Schlüssels zu überprüfen.

```
gpg --fingerprint 7EEF030EDDF5C21C
```

Der Befehl sollte einen Fingerabdruck zurückgeben, der mit dem folgenden identisch ist:

```
1C24 32C1 862F 64D1 F90A  239A 7EEF 030E DDF5 C21C
```

> **Important**
>
> Führen Sie das Slurm-Installationsskript nicht aus, wenn der Fingerabdruck nicht übereinstimmt. AWS Support kontaktieren.

c. Laden Sie die Signaturdatei herunter und überprüfen Sie die Signatur der Tarball-Datei des Slurm-Installationsprogramms. *region*Ersetzen Sie durch den AWS-Region Ort, an dem Sie Ihre temporäre Instanz gestartet haben, z. B. `us-east-1`

```
wget https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-25.11-installer-25.11.2-1.tar.gz.sig && \
     gpg --verify ./aws-pcs-slurm-25.11-installer-25.11.2-1.tar.gz.sig
```

Die Ausgabe sollte folgendermaßen oder ähnlich aussehen:

```
gpg: assuming signed data in './aws-pcs-slurm-25.11-installer-25.11.2-1.tar.gz'
gpg: Signature made Thu 26 Mar 2026 08:57:11 AM UTC using RSA key ID ECC0AE5C
gpg: Good signature from "AWS PCS Packages (AWS PCS Packages)"
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
```

```
Primary key fingerprint: 1C24 32C1 862F 64D1 F90A  239A 7EEF 030E DDF5 C21C
     Subkey fingerprint: B7E1 8788 3517 6A74 C3D5  EAF5 6088 136D ECC0 AE5C
```

Wenn das Ergebnis den Fingerabdruck enthält `Good signature` und der Fingerabdruck mit dem im vorherigen Schritt zurückgegebenen Fingerabdruck übereinstimmt, fahren Sie mit dem nächsten Schritt fort.

> **Important**
>
> Führen Sie das Slurm-Installationsskript nicht aus, wenn der Fingerabdruck nicht übereinstimmt. AWS Support kontaktieren.

4. Extrahieren Sie die Daten aus der komprimierten `.tar.gz`-Datei und wechseln Sie in das extrahierte Verzeichnis.

```
tar -xf aws-pcs-slurm-25.11-installer-25.11.2-1.tar.gz && \
    cd aws-pcs-slurm-25.11-installer
```

5. Installieren Sie Slurm. Das Installationsprogramm lädt Slurm und seine Abhängigkeiten herunter, kompiliert und installiert sie. Es dauert mehrere Minuten, abhängig von den Spezifikationen der ausgewählten temporären Instanz.

```
sudo ./installer.sh -y
```

6. Überprüfen Sie die Scheduler-Versionsdatei, um die Installation zu bestätigen.

```
cat /opt/aws/pcs/scheduler/slurm-25.11/version
```

Die Ausgabe sollte folgendermaßen oder ähnlich aussehen:

```
SLURM_INSTALL_DATE='Thu Mar 26 15:15:37 UTC 2026'
SLURM_VERSION='25.11.2'
PCS_SLURM_RELEASE='1'
```

Schritt 4 — (Optional) Zusätzliche Treiber, Bibliotheken und Anwendungssoftware installieren

Installieren Sie zusätzliche Treiber, Bibliotheken und Anwendungssoftware auf der temporären Instanz. Die Installationsverfahren variieren je nach den spezifischen Anwendungen und Bibliotheken. Wenn Sie noch kein benutzerdefiniertes AMI für AWS PCS erstellt haben, empfehlen wir Ihnen, zunächst ein AMI nur mit der AWS PCS-Software und installiertem Slurm zu erstellen und zu testen und dann schrittweise Ihre eigene Software und Konfigurationen hinzuzufügen, sobald Sie den ersten Erfolg bestätigt haben.

Beispiele

- Software für Elastic Fabric Adapter (EFA). Weitere Informationen finden Sie unter Erste Schritte mit EFA und MPI für HPC-Workloads auf Amazon EC2 im Amazon Elastic Compute Cloud-Benutzerhandbuch.
- Client für Amazon Elastic File System (Amazon EFS). Weitere Informationen finden Sie unter Manuelles Installieren des Amazon EFS-Clients im Amazon Elastic File System-Benutzerhandbuch.
- Lustre-Client, um Amazon FSx for Lustre und Amazon File Cache zu verwenden. Weitere Informationen finden Sie unter Installation des Lustre-Clients im FSx for Lustre-Benutzerhandbuch.
- CloudWatch Amazon-Agent, um CloudWatch Logs and Metrics zu verwenden. Weitere Informationen finden Sie unter Installieren des CloudWatch Agenten im CloudWatch Amazon-Benutzerhandbuch.
- AWS Neuron, um die Instance-Typen trn* und inf* zu verwenden. Weitere Informationen finden Sie in der Neuron-Dokumentation.AWS
- NVIDIA-Treiber, CUDA und DCGM, um die Instanztypen p* oder g* zu verwenden.

Schritt 5 — Erstellen Sie ein mit AWS PCS kompatibles AMI

Nachdem Sie die erforderlichen Softwarekomponenten installiert haben, erstellen Sie ein AMI, das Sie wiederverwenden können, um Instances in AWS PCS-Compute-Knotengruppen zu starten.

> **Important**
>
> AWS PCS benötigt derzeit einen Kernel mit IPv4 Unterstützung für lokale Knotenkommunikation, auch wenn Sie AWS PCS in einem IPv6 reinen Netzwerk verwenden.

So erstellen Sie ein AMI aus Ihrer temporären Instance:

1. Öffnen Sie die Amazon EC2-Konsole.
2. Wählen Sie im Navigationsbereich Instances aus.
3. Wählen Sie die temporäre Instanz aus, die Sie erstellt haben. Wählen Sie Aktionen, Image, Image erstellen.
4. Gehen Sie bei Create Image (Image erstellen) wie folgt vor:
   a. Geben Sie unter Image name (Image-Name) einen beschreibenden Namen für das AMI ein.
   b. (Optional:) Geben Sie bei Image description (Image-Beschreibung) eine kurze Beschreibung des Zwecks des AMI ein.
   c. Wählen Sie Create Image (Image erstellen) aus.
5. Wählen Sie im Navigationsbereich AMIs aus.
6. Suchen Sie das AMI, das Sie erstellt haben, in der Liste. Warten Sie, bis sich der Status von Ausstehend auf Verfügbar ändert, und verwenden Sie es dann mit einer AWS PCS-Compute-Knotengruppe.

Schritt 6 — Verwenden Sie das benutzerdefinierte AMI mit einer AWS PCS-Compute-Knotengruppe

Sie können Ihr benutzerdefiniertes AMI mit einer neuen oder vorhandenen AWS PCS-Compute-Knotengruppe verwenden.

> **Important**
>
> AWS PCS benötigt derzeit einen Kernel mit IPv4 Unterstützung für lokale Knotenkommunikation, auch wenn Sie AWS PCS in einem IPv6 reinen Netzwerk verwenden.

New compute node group

Um das benutzerdefinierte AMI zu verwenden

1. Öffnen Sie die AWS PCS-Konsole.
2. Klicken Sie im Navigationsbereich auf Cluster.

3. Wählen Sie den Cluster aus, in dem Sie das benutzerdefinierte AMI verwenden möchten, und wählen Sie dann Compute Node Groups aus.
4. Erstellen Sie eine neue Compute-Knotengruppe. Weitere Informationen finden Sie unter Erstellen einer Compute-Knotengruppe in AWS STK.. Suchen Sie unter AMI-ID nach dem Namen oder der ID des benutzerdefinierten AMI, das Sie verwenden möchten. Schließen Sie die Konfiguration der Compute-Knotengruppe ab und wählen Sie dann Create Compute Node Group aus.
5. (Optional) Vergewissern Sie sich, dass das AMI Instance-Starts unterstützt. Starten Sie eine Instance in der Compute-Knotengruppe. Sie können dies tun, indem Sie die Compute-Knotengruppe so konfigurieren, dass sie über eine einzelne statische Instanz verfügt, oder Sie können einen Job an eine Warteschlange senden, die die Compute-Knotengruppe verwendet.

    a. Überprüfen Sie die Amazon EC2 EC2-Konsole, bis eine Instance angezeigt wird, die mit der neuen Compute-Knotengruppen-ID gekennzeichnet ist. Weitere Informationen dazu finden Sie unter.. Suchen nach Rechenknotengruppeninstanzen in AWS PCS
    b. Wenn Sie sehen, dass eine Instance gestartet wird und ihr Bootstrap-Vorgang abgeschlossen ist, vergewissern Sie sich, dass sie das erwartete AMI verwendet. Wählen Sie dazu die Instance aus und überprüfen Sie dann die AMI-ID unter Details. Es sollte dem AMI entsprechen, das Sie in den Einstellungen der Compute-Knotengruppe konfiguriert haben.
    c. (Optional) Aktualisieren Sie die Skalierungskonfiguration für die Compute-Knotengruppe auf Ihre bevorzugten Werte.

Existing compute node group

Um das benutzerdefinierte AMI zu verwenden

1. Öffnen Sie die AWS PCS-Konsole.
2. Klicken Sie im Navigationsbereich auf Cluster.
3. Wählen Sie den Cluster aus, in dem Sie das benutzerdefinierte AMI verwenden möchten, und wählen Sie dann Compute Node Groups aus.
4. Wählen Sie die Knotengruppe aus, die Sie konfigurieren möchten, und klicken Sie auf Bearbeiten. Suchen Sie unter AMI-ID nach dem Namen oder der ID des benutzerdefinierten AMI, das Sie verwenden möchten. Beenden Sie die Konfiguration der Compute-Knotengruppe und wählen Sie dann Update. Neue Instances, die in der Compute-

Knotengruppe gestartet werden, verwenden die aktualisierte AMI-ID. Bestehende Instances werden weiterhin das alte AMI verwenden, bis AWS PCS sie ersetzt. Weitere Informationen finden Sie unter Aktualisierung eines AWS PCS-Compute-Knotengruppe.

5. (Optional) Vergewissern Sie sich, dass das AMI Instance-Starts unterstützt. Starten Sie eine Instance in der Compute-Knotengruppe. Sie können dies tun, indem Sie die Compute-Knotengruppe so konfigurieren, dass sie über eine einzelne statische Instanz verfügt, oder Sie können einen Job an eine Warteschlange senden, die die Compute-Knotengruppe verwendet.

   a. Überprüfen Sie die Amazon EC2 EC2-Konsole, bis eine Instance angezeigt wird, die mit der neuen Compute-Knotengruppen-ID gekennzeichnet ist. Weitere Informationen dazu finden Sie unter.. Suchen nach Rechenknotengruppeninstanzen in AWS PCS

   b. Wenn Sie sehen, dass eine Instance gestartet wird und ihr Bootstrap-Vorgang abgeschlossen ist, vergewissern Sie sich, dass sie das erwartete AMI verwendet. Wählen Sie dazu die Instance aus und überprüfen Sie dann die AMI-ID unter Details. Es sollte dem AMI entsprechen, das Sie in den Einstellungen der Compute-Knotengruppe konfiguriert haben.

   c. (Optional) Aktualisieren Sie die Skalierungskonfiguration für die Compute-Knotengruppe auf Ihre bevorzugten Werte.

Schritt 7 — Beenden Sie die temporäre Instanz

Nachdem Sie bestätigt haben, dass Ihr AMI wie vorgesehen mit AWS PCS funktioniert, können Sie die temporäre Instance beenden, damit keine Gebühren mehr dafür anfallen.

So beenden Sie die temporäre Instance:

1. Öffnen Sie die Amazon EC2-Konsole.
2. Wählen Sie im Navigationsbereich Instances aus.
3. Wählen Sie die temporäre Instance aus, die Sie erstellt haben, und wählen Sie Actions, Instance state, Terminate Instance aus.
4. Wenn Sie zur Bestätigung aufgefordert werden, wählen Sie Terminate.

Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS

AWS stellt eine herunterladbare Datei bereit, mit der die AWS PCS-Software auf einer Instance installiert werden kann. AWS stellt auch Software bereit, mit der relevante Versionen von Slurm und seinen Abhängigkeiten heruntergeladen, kompiliert und installiert werden können. Sie können diese Anweisungen verwenden, um benutzerdefinierte AMIs für die Verwendung mit AWS PCS zu erstellen, oder Sie können Ihre eigenen Methoden verwenden.

Inhalt

AWS Installationsprogramm für die PCS-Agentensoftware

Das AWS PCS-Agent-Softwareinstallationsprogramm konfiguriert eine Instanz so, dass sie während des Instanz-Bootstrap-Vorgangs mit AWS PCS zusammenarbeitet. Sie müssen die von AWS-bereitgestellten Installationsprogramme verwenden, um den AWS PCS-Agenten auf Ihrem benutzerdefinierten AMI zu installieren.

Weitere Informationen zur AWS PCS-Agent-Software finden Sie unter. AWS Versionen von PCS-Agenten

Slurm-Installationsprogramm

Das Slurm-Installationsprogramm lädt relevante Versionen von Slurm und seinen Abhängigkeiten herunter, kompiliert und installiert sie. Sie können das Slurm-Installationsprogramm verwenden, um benutzerdefinierte AMIs für PCS zu erstellen. AWS Sie können auch Ihre eigenen Mechanismen verwenden, sofern diese mit der Softwarekonfiguration übereinstimmen, die der Slurm-Installer bereitstellt. Weitere Informationen zur AWS PCS-Unterstützung für Slurm finden Sie unter. Slurm-Versionen in AWS STK.

Die AWS mitgelieferte Software installiert Folgendes:

- Slurm auf der angeforderten Haupt- und Wartungsversion (derzeit Version 25.11.x) — Lizenz GPL 2
  - Slurm wurde mit folgender Einstellung gebaut `--sysconfdir /etc/slurm`
  - Slurm wurde mit der Option gebaut und `--enable-pam --without-munge`
  - Slurm wurde mit der Option gebaut `--sharedstatedir=/run/slurm/`
  - Slurm wurde mit PMIX- und JWT-Unterstützung erstellt
  - Slurm ist installiert unter `/opt/aws/pcs/schedulers/slurm-25.11`
- OpenPMix (Version 4.2.6) — Lizenz
  - OpenPMix ist als Unterverzeichnis installiert von `/opt/aws/pcs/scheduler/`
- libjwt (Version 1.17.0) — Lizenz MPL-2.0
  - libjwt ist als Unterverzeichnis installiert von `/opt/aws/pcs/scheduler/`

Die AWS mitgelieferte Software ändert die Systemkonfiguration wie folgt:

- Die durch den Build erstellte `systemd` Slurm-Datei wird `/etc/systemd/system/` mit dem Dateinamen kopiert. `slurmd-25.11.service`
- Falls sie nicht existieren, werden ein Slurm-Benutzer und eine Gruppe (`slurm:slurm`) mit UID/GID of erstellt. `401`
- Der Ordner `/etc/aws/pcs/scheduler/slurm-25.11/plugstack.conf.d/` wird erstellt, um Ihre Erweitern Sie die Slurm-Funktionalität auf AWS PCS mit SPANK-Plugins Konfiguration zu speichern.
- Auf Amazon Linux 2 und Rocky Linux 9 fügt die Installation das EPEL Repository hinzu, um die erforderliche Software zur Erstellung von Slurm oder seinen Abhängigkeiten zu installieren.
- Auf RHEL9 ermöglicht die Installation `codeready-builder-for-rhel-9-rhui-rpms` und `epel-release-latest-9` von die Installation der erforderlichen Software `fedoraproject` zum Erstellen von Slurm oder seinen Abhängigkeiten.

Unterstützte Betriebssysteme

Siehe Unterstützte Betriebssysteme in AWS PCS.

> **Note**
>
> AWS Deep Learning AMIs (DLAMI) -Versionen, die auf Amazon Linux 2023 und Ubuntu 22.04 basieren, sollten mit der AWS PCS-Software und den Slurm-Installationsprogrammen kompatibel sein. Weitere Informationen finden Sie unter Choosing Your DLAMI im AWS Deep Learning AMIs Developer Guide.

Unterstützte Instance-Typen

AWS PCS-Software und Slurm-Installationsprogramme unterstützen jeden x86_64- oder arm64-Instanztyp, auf dem eines der unterstützten Betriebssysteme ausgeführt werden kann.

Unterstützte Slurm-Versionen

Siehe Slurm-Versionen in AWS STK..

Überprüfen Sie die Installationsprogramme anhand einer Prüfsumme

Sie können SHA256-Prüfsummen verwenden, um die Tarball-Dateien (.tar.gz) des Installers zu überprüfen. Diese Vorgehensweise wird empfohlen, um die Identität des Software-Publishers zu überprüfen und zu prüfen, ob die Anwendung seit der Veröffentlichung nicht verändert oder beschädigt wurde.

Um einen Tarball zu verifizieren

Verwenden Sie das Hilfsprogramm sha256sum für die SHA256-Prüfsumme und geben Sie den Tarball-Dateinamen an. Sie müssen den Befehl von dem Verzeichnis aus ausführen, in dem Sie die Tarball-Datei gespeichert haben.

- SHA256

```
$ sha256sum tarball_filename.tar.gz
```

Der Befehl sollte einen Prüfsummenwert im folgenden Format zurückgeben.

```
checksum_value tarball_filename.tar.gz
```

Vergleichen Sie den vom Befehl zurückgegebenen Prüfsummenwert mit dem in der folgenden Tabelle angegebenen Prüfsummenwert. Wenn die Prüfsummen übereinstimmen, ist es sicher, das Installationsskript auszuführen.

> **Important**
>
> Wenn die Prüfsummen nicht übereinstimmen, führen Sie das Installationsskript nicht aus. Wenden Sie sich an Support.

Der folgende Befehl generiert beispielsweise die SHA256-Prüfsumme für den Slurm 25.11.2-1-Tarball.

```
$ sha256sum aws-pcs-slurm-25.11-installer-25.11.2-1.tar.gz
```

Beispielausgabe:

```
aa063bc01b2ccd84a82402e8b8dbcd8c7401ebd2e0a670c867d77167944d621a aws-pcs-slurm-25.11-installer-25.11.2-1.tar.gz
```

In den folgenden Tabellen sind die Prüfsummen für die neuesten Versionen der Installationsprogramme aufgeführt. Ersetzen Sie es *us-east-1* durch das AWS-Region , wo Sie PCS verwenden AWS .

AWS PCS-Agent

Installer (Installationsprogramm)	URL herunterladen	SHA256-Prüfsumme
AWS PCS-Agent 1.4.0-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.4.0-1.tar.gz`	`e9a342478483df8e4666741ef2aad504321676508d489f88e96311297990a17f`
AWS PCS-Agent 1.3.2-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/`	`06b32a952a1c849e3442e35c28ac2e4d6962`

Installer (Installationsprogramm)	URL herunterladen	SHA256-Prüfsumme
	aws-pcs-agent/aws-pcs-agent-v1.3.2-1.tar.gz	b09286cad748f3c83d561b52ec6f
AWS PCS-Agent 1.3.1-1	https://aws-pcs-repo-*us-east-1*.s3.*us-east-1*.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.3.1-1.tar.gz	5b7f1eb7b3a86bd2d331b5cb0138d868dc9452da34b480becd86af892c7e8d19
AWS PCS-Agent 1.3.0-1	https://aws-pcs-repo-*us-east-1*.s3.*us-east-1*.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.3.0-1.tar.gz	eadc9b65c3db248bdde2a6c41814dfb1b97239f24ad55e03d8526dd9ab4a8d16
AWS PCS-Agent 1.2.2-1	https://aws-pcs-repo-*us-east-1*.s3.*us-east-1*.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.2.2-1.tar.gz	fd7b6ea5442db75d723fc4971781ce6ae511baa21b87c4286fc1df8127b282b8
AWS PCS-Agent 1.2.1-1	https://aws-pcs-repo-*us-east-1*.s3.*us-east-1*.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.2.1-1.tar.gz	2b784643ca01ccca1baa64fbfb34bb41efe8bdca69470998b74ce3962bc271d4

Installer (Installationsprogramm)	URL herunterladen	SHA256-Prüfsumme
AWS PCS-Agent 1.2.0-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.2.0-1.tar.gz`	`470db8c4fc9e50277b6317f98584b6b547e73523043e34f018eeca e767846805`
AWS PCS-Agent 1.1.1-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.1.1-1.tar.gz`	`bef078bf60a6d8ecde2e6c49cd34d088703f02550279e3bf483d57a235334dc6`
AWS PCS-Agent 1.1.0-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.1.0-1.tar.gz`	`594c32194c71bccc5d66e5213213ae38dd2c6d2f9a950bb01accea 0bbab0873a`
AWS PCS-Agent 1.0.1-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.0.1-1.tar.gz`	`04e22264019837e3f42d8346daf5886eaace cd21571742eb505ea8911786bcb2`
AWS PCS-Agent 1.0.0-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.0.0-1.tar.gz`	`d2d3d68d00c685435c38af471d7e2492dde5ce9eb222d7b6ef0042 144b134ce0`

Slurm-Installationsprogramm

Installer (Installationsprogramm)	URL herunterladen	SHA256-Prüfsumme
Slurm 25.11.2-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-25.11-installer-25.11.2-1.tar.gz`	`aa063bc01b2ccd84a82402e8b8dbcd8c7401ebd2e0a670c867d77167944d621a`
Slurm 25.05.7-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-25.05-installer-25.05.7-1.tar.gz`	`5019436389649ce0cacf04cd1d1adf1a4e46b9291967af7bf5f0a8ac4a49e4f0`
Slurm 25.05.5-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-25.05-installer-25.05.5-1.tar.gz`	`e7bc84db4e71b8c7174e2f581a31233f839affb5306c76a8adba23204dcc703b`
Slurm 25.05.4-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-25.05-installer-25.05.4-1.tar.gz`	`3b0f93bce441d4f4f6935175f2c1e81cd961cb923adb416fa6689f5592047a7d`
Slurm 25.05.3-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs`	`851bb5815b6700ceb30cc4a3fda204ca8ce362c14528c339908983255a936cf0`

Installer (Installationsprogramm)	URL herunterladen	SHA256-Prüfsumme
	-slurm-25.05-installer-25.05.3-1.tar.gz	
Slurm 24.11.7-1	https://aws-pcs-repo- *us-east-1* .s3.*us-east-1* .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.11-installer-24.11.7-1.tar.gz	73d75be82c6f88f6e248fd0cc779a5630c62d91ebabdd9cf0f61b1943b6d7d09
Slurm 24.11.6-2	https://aws-pcs-repo- *us-east-1* .s3.*us-east-1* .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.11-installer-24.11.6-2.tar.gz	f17cd78e0bc6b9c818b794d9d2685cceabdc73f4fbb12f7566ae5b86a5abc32b
Slurm 24.11.6-1	https://aws-pcs-repo- *us-east-1* .s3.*us-east-1* .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.11-installer-24.11.6-1.tar.gz	225de9fc18206f5f65f412effe1fd457614ac97ee9822b3ff804a452b0fae522
Slurm 24.11.5-1	https://aws-pcs-repo- *us-east-1* .s3.*us-east-1* .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.11-installer-24.11.5-1.tar.gz	593efe4d66bef2f3e46d5a382fb5a32f7a3ca2510bcf1b3c85739f4f951810d5

Installer (Installationsprogramm)	URL herunterladen	SHA256-Prüfsumme
Slurm 24.05.8-2	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.05-installer-24.05.8-2.tar.gz`	`c494b0b55c319a4c2f3faf668c759d46c32c4c7aa94ae97d94128328fe95364b`
Slurm 24.05.8-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.05-installer-24.05.8-1.tar.gz`	`210a43b376af082bbad640b2032655885790c5dab0e6489cc327c7310a375849`
Slurm 24.05.7-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.05-installer-24.05.7-1.tar.gz`	`0b5ed7c81195de2628c78f37c79e63fc4ae99132ca6b019b53a0d68792ee82c5`
Slurm 24.05.5-2	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.05-installer-24.05.5-2.tar.gz`	`7cc8d8294f2fbff95fe0602cf9e21e02003b5d96c0730e0a18c6aa04c7a4967b`
Slurm 23.11.10-4 (veraltet)	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.10-4.tar.gz`	`bb2d8c919c69dba38d14358f49c7f0427564c5dd4af85a1c9eca2c57ceeae29a`

Installer (Installationsprogramm)	URL herunterladen	SHA256-Prüfsumme
Slurm 23.11.10-3 (veraltet)	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.10-3.tar.gz`	`488a10ee0fbd57ec0e0ff7ea708a9e3038fafdc025c6bb391c75c2e2a7852a00`
Slurm 23.11.10-2 (veraltet)	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.10-2.tar.gz`	`0bbe85423305c05987931168caf98da08a34c25f9eec0690e8e74de0b7bc8752`
Slurm 23.11.10-1 (veraltet)	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.10-1.tar.gz`	`27e8faa9980e92cdfd8cfdc71f937777f0934552ce61e33dac4ecf5a20321e44`
Slurm 23.11.9-1 (veraltet)	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz`	`1de7d919c8632fe8e2806611bed4fde1005a4fadc795412456e935c7bba2a9b8`

Versionshinweise für AWS PCS-Beispiel-AMIs

AMIs für die neuesten unterstützten Hauptversionen des Schedulers erhalten Sicherheitsupdates und wichtige Bugfixes. Diese inkrementellen Sicherheitspatches sind nicht in den offiziellen Versionshinweisen enthalten.

> **Important**
>
> Beispiel-AMIs, die sich auf alte Scheduler-Versionen beziehen, werden nicht unterstützt und erhalten keine Updates.

> **Important**
>
> Beispiel-AMIs dienen zu Demonstrationszwecken und werden nicht für Produktionsworkloads empfohlen.

Inhalt

AWS PCS-Beispiel-AMIs für x86_64

Slurm 25.11

AMI-Name

- `aws-pcs-sample_ami-al2023-x86_64-slurm-25.11`

Unterstützte EC2-Instances

- Alle Instanzen mit einem 64-Bit-x86-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur Amazon EC2 EC2-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach Architectures=x86_64.

AMI-Inhalte

- Unterstützter AWS-Service: AWS PCS
- Betriebssystem: Amazon Linux 2023
- Rechenarchitektur: x86_64
- EBS-Volumetyp: gp2
- EFA-Installationsprogramm: 1.47.0
- GDR-Kopie: 2.5.1
- NVIDIA-Treiber: 590.48.01
- NVIDIA CUDA: 13.1_590.48.01

Slurm 25.05

AMI-Name

- `aws-pcs-sample_ami-amzn2-x86_64-slurm-25.05`

Unterstützte EC2-Instances

- Alle Instanzen mit einem 64-Bit-x86-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur Amazon EC2 EC2-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach Architectures=x86_64.

AMI-Inhalte

- Unterstützter AWS-Service: AWS PCS
- Betriebssystem: Amazon Linux 2
- Rechenarchitektur: x86_64
- EBS-Volumetyp: gp2
- EFA-Installationsprogramm: 1.43.1
- GDR-Kopie: 2.5.1
- NVIDIA-Treiber: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Slurm 24.11

> **Note**
>
> AWS PCS unterstützt die Buchhaltung für Slurm 24.11 und höher. Weitere Informationen finden Sie unter Slurm-Buchhaltung in AWS STK..

AMI-Name

- `aws-pcs-sample_ami-amzn2-x86_64-slurm-24.11`

Unterstützte EC2-Instances

- Alle Instanzen mit einem 64-Bit-x86-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur Amazon EC2 EC2-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach`Architectures=x86_64`.

AMI-Inhalte

- Unterstützter AWS Dienst: AWS PCS
- Betriebssystem: Amazon Linux 2
- Rechenarchitektur: x86_64
- EBS-Volumetyp: gp2
- EFA-Installationsprogramm: 1.33.0
- GDRCopy:: 2.4
- NVIDIA-Treiber: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Slum 24.05

AMI-Name

- `aws-pcs-sample_ami-amzn2-x86_64-slurm-24.05`

Unterstützte EC2-Instances

- Alle Instanzen mit einem 64-Bit-x86-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur Amazon EC2 EC2-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach`Architectures=x86_64`.

AMI-Inhalte

- Unterstützter AWS Dienst: AWS PCS
- Betriebssystem: Amazon Linux 2
- Rechenarchitektur: x86_64
- EBS-Volumetyp: gp2
- EFA-Installationsprogramm: 1.33.0
- GDRCopy:: 2.4
- NVIDIA-Treiber: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Slurm 23.11

AMI-Name

- `aws-pcs-sample_ami-amzn2-x86_64-slurm-23.11`

Unterstützte EC2-Instances

- Alle Instanzen mit einem 64-Bit-x86-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur Amazon EC2 EC2-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach`Architectures=x86_64`.

AMI-Inhalte

- Unterstützter AWS Dienst: AWS PCS
- Betriebssystem: Amazon Linux 2
- Rechenarchitektur: x86_64
- EBS-Volumetyp: gp2

- EFA-Installationsprogramm: 1.33.0
- GDRCopy:: 2.4
- NVIDIA-Treiber: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

AWS PCS-Beispiel-AMIs für Arm64

Slurm 25.11

AMI-Name

- `aws-pcs-sample_ami-al2023-arm64-slurm-25.11`

Unterstützte EC2-Instances

- Alle Instances mit einem 64-Bit-ARM-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur Amazon EC2 EC2-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach Architectures=ARM64.

AMI-Inhalte

- Unterstützter AWS-Service: AWS PCS
- Betriebssystem: Amazon Linux 2023
- Rechenarchitektur: arm64
- EBS-Volumetyp: gp2
- EFA-Installationsprogramm: 1.47.0
- GDR-Kopie: 2.5.1
- NVIDIA-Treiber: 590.48.01
- NVIDIA CUDA: 13.1_590.48.01

Slurm 25.05

AMI-Name

- `aws-pcs-sample_ami-amzn2-arm64-slurm-25.05`

Unterstützte EC2-Instances

- Alle Instances mit einem 64-Bit-ARM-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur Amazon EC2 EC2-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach Architectures=ARM64.

AMI-Inhalte

- Unterstützter AWS-Service: AWS PCS
- Betriebssystem: Amazon Linux 2
- Rechenarchitektur: arm64
- EBS-Volumetyp: gp2
- EFA-Installationsprogramm: 1.43.1
- GDR-Kopie: 2.5.1
- NVIDIA-Treiber: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Slurm 24.11

> **Note**
>
> AWS PCS unterstützt die Buchhaltung für Slurm 24.11 und höher. Weitere Informationen finden Sie unter Slurm-Buchhaltung in AWS STK..

AMI-Name

- `aws-pcs-sample_ami-amzn2-arm64-slurm-24.11`

Unterstützte EC2-Instances

- Alle Instances mit einem 64-Bit-ARM-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur Amazon EC2 EC2-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach`Architectures=arm64`.

AMI-Inhalte

- Unterstützter AWS Dienst: AWS PCS
- Betriebssystem: Amazon Linux 2
- Rechenarchitektur: arm64
- EBS-Volumetyp: gp2
- EFA-Installationsprogramm: 1.33.0
- GDRCopy:: 2.4
- NVIDIA-Treiber: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Slum 24.05

AMI-Name

- `aws-pcs-sample_ami-amzn2-arm64-slurm-24.05`

Unterstützte EC2-Instances

- Alle Instances mit einem 64-Bit-ARM-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur Amazon EC2 EC2-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach`Architectures=arm64`.

AMI-Inhalte

- Unterstützter AWS Dienst: AWS PCS
- Betriebssystem: Amazon Linux 2
- Rechenarchitektur: arm64
- EBS-Volumetyp: gp2
- EFA-Installationsprogramm: 1.33.0
- GDRCopy:: 2.4
- NVIDIA-Treiber: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Slurm 23.11

AMI-Name

- `aws-pcs-sample_ami-amzn2-arm64-slurm-23.11`

Unterstützte EC2-Instances

- Alle Instances mit einem 64-Bit-ARM-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur Amazon EC2 EC2-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach`Architectures=arm64`.

AMI-Inhalte

- Unterstützter AWS Dienst: AWS PCS
- Betriebssystem: Amazon Linux 2
- Rechenarchitektur: arm64
- EBS-Volumetyp: gp2
- EFA-Installationsprogramm: 1.33.0
- GDRCopy:: 2.4
- NVIDIA-Treiber: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Unterstützte Betriebssysteme in AWS PCS

AWS PCS verwendet das für eine Rechenknotengruppe konfigurierte Amazon Machine Image (AMI), um EC2-Instances in dieser Rechenknotengruppe zu starten. Das AMI bestimmt das Betriebssystem, das die EC2-Instances verwenden. Sie können das Betriebssystem in AWS PCS-Beispiel-AMIs nicht ändern. Sie müssen ein benutzerdefiniertes AMI erstellen, wenn Sie ein anderes Betriebssystem verwenden möchten. Weitere Informationen finden Sie unter Amazon Machine Images (AMIs) für AWS STK..

Unterstützte Betriebssysteme

- Amazon Linux 2023

  Dies ist das Betriebssystem in den AWS PCS-Beispiel-AMIs.

  > **Important**
  >
  > Beispiel-AMIs dienen zu Demonstrationszwecken und werden nicht für Produktionsworkloads empfohlen. Sie sollten ein benutzerdefiniertes AMI für Produktionsworkloads erstellen und verwenden, auch wenn Sie Amazon Linux 2023 verwenden möchten.

- Amazon Linux 2
- RedHat Linux 9 für Unternehmen (RHEL 9)

  Die On-Demand-Kosten für RHEL sind für jeden Instanztyp höher als für andere unterstützte Betriebssysteme. Weitere Informationen zu den Preisen finden Sie unter On-Demand Preise und Wie wird Red Hat Enterprise Linux auf Amazon Elastic Compute Cloud angeboten und wie wird der Preis berechnet? .

- RedHat Linux 8 für Unternehmen (RHEL 8)
- Rocky Linux 9

  Sie können die offiziellen Rocky Linux 9-AMIs als Basis für ein benutzerdefiniertes AMI verwenden. Ihr benutzerdefinierter AMI-Build schlägt möglicherweise fehl, wenn das Basis-AMI nicht über den neuesten Kernel verfügt.

Um den Kernel zu aktualisieren

1. Starten Sie von hier aus eine Instance mit einer Rocky9-AMI-ID: https://rockylinux.org/cloud-images/
2. Rufen Sie die Instance per SSH auf und führen Sie den folgenden Befehl aus:

```
sudo yum -y update
```

3. Erstellen Sie ein Bild von der Instanz. Sie geben dieses Image als das `ParentImage` für Ihr benutzerdefiniertes AMI an.

- Rocky Linux 8
- Ubuntu 22.04

  Ubuntu 22.04 benötigt sicherere Schlüssel für SSH und unterstützt standardmäßig keine RSA-Schlüssel. Wir empfehlen Ihnen, stattdessen einen ED25519-Schlüssel zu generieren und zu verwenden.
- Ubuntu 24.04

AWS Versionen von PCS-Agenten

Die AWS PCS-Agentensoftware konfiguriert die EC2-Instances, die AWS PCS startet, für die Verwendung mit Slurm. Sie nehmen den Agenten in ein Amazon Machine Images (AMI) auf, das Sie angeben, wenn Sie Rechenknotengruppen für Ihren Cluster erstellen. Die in diesen Compute-Knotengruppen gestarteten EC2-Instances verwenden das angegebene AMI und die darin enthaltene AWS PCS-Agent-Software. Der AWS PCS-Agent ermöglicht es einer EC2-Instance, sich selbst als Teil des Clusters zu registrieren. Um die neueste AWS PCS-Agent-Software verwenden zu können, müssen Sie Ihre benutzerdefinierten AMIs aktualisieren. Weitere Informationen finden Sie unter Schritt 2 — Installieren Sie den AWS PCS-Agenten in Benutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS.

AWS Version des PCS-Agenten	Datum der Veröffentlichung	Versionshinweise
v1.4.0-1	7. Mai 2026	• Unterstützung für die Konfiguration und den Start des `slurmd` Daemons für jede Slurm-Version hinzugefügt, die mit dem Slurm-Controller kompatibel ist, der auf dem PCS-Cluster läuft. AWS
v1.3.2-1	10. März 2026	• Es wurde ein Problem behoben, bei dem Rechenknoten, auf denen RHEL 8.10 oder Rocky Linux 8.10 ausgeführt wurden, aufgrund eines fehlerhaften `curl` SigV4-Backports in diesen Betriebssystemen kein Bootstrap durchgeführt werden konnten.

AWS Version des PCS-Agenten	Datum der Veröffentlichung	Versionshinweise
v1.3.1-1	7. November 2025	• Verbesserte Deaktivierung von Hyperthreading durch Verwendung des smt/control ``sysfs-Parameters, sofern verfügbar. • Es wurde ein mögliches Race-Problem behoben, bei dem die CPU während des Startvorgangs gesperrt ist, während der PCS-Agent versucht, Hyperthreading zu deaktivieren. • Es wurde ein Problem behoben, das dazu führte, dass die `InstanceType` Felder `InstanceId` und der Slurm-Rechenknoten mit einem Zeitstempel bzw. einem Bindestrich gefüllt wurden.
v1.3.0-1	3. November 2025	• Unterstützung für neue Betriebssysteme hinzugefügt: Amazon Linux 2023, Ubuntu 24, RHEL 8, Rocky 8.

AWS Version des PCS-Agenten	Datum der Veröffentlichung	Versionshinweise
v1.2.2-1	16. Oktober 2025	• Zulässige Abfragen von Instanz-Metadaten an einen IPv6-Endpunkt, wenn ein IPv4-Endpunkt nicht verfügbar ist. • Es wurde ein Problem behoben, das die Deaktivierung von Hyperthreading verhinderte, wenn der Kernel gleichgeordnete Threads als CPU-ID-Bereiche zurückgab. • Es wurde ein Problem behoben, das zu falschen Fehlermeldungen in den Protokollen führte, wenn Hyperthreading erfolgreich deaktiviert wurde.
v1.2.1-1	19. Juni 2025	• Der AWS PCS-Agent versucht nun für bis zu 30 Minuten, slurmd zu starten, wenn der Controller nicht verfügbar ist. • Es wurde ein Problem behoben, das zu einer falschen Slurmd-Konfiguration führte, wenn die Antwort auf einen SLURMDBD-Endpunkt RegisterComputeNodeGroupInstance enthielt.

AWS Version des PCS-Agenten	Datum der Veröffentlichung	Versionshinweise
v1.2.0-1	7. März 2025	• Aktiviert die Unterstützung für IPv6 in. `slurmd.conf`
v1.1.1-1	13. Dezember 2024	• Es wurde ein Problem behoben, bei dem im Call to RegisterComputeNodeGroupInstance eine falsche Slurm-Version gemeldet wurde. • Es wurde ein Problem behoben, bei dem Instanz-Metadaten nicht korrekt abgerufen wurden, wenn ein benutzerdefiniertes Skript ausgeführt `/opt/aws/pcs/etc/bootstrap_hooks/` wurde.
v1.1.0-1	6. Dezember 2024	• Benutzerdefinierte Skripts wurden aktiviert, damit sie vor dem `/opt/aws/pcs/etc/bootstrap_hooks/` Bootstrap-Schritt ausgeführt werden können.
v1.0.1-1	22. Oktober 2024	• Es wurde ein Problem behoben, bei dem NVIDIA-Geräte nicht funktionierten, wenn sie auf GPU-enabled Instanzen `slurmd` gestartet wurden.
v1.0.0-1	28. August 2024	• Erstversion.

Slurm-Scheduler einschalten AWS STK.

Slurm ist ein Open-Source-Workload-Manager, der für Linux-Cluster entwickelt wurde und Funktionen zur Jobplanung, Ressourcenzuweisung und Jobüberwachung für HPC-Workloads bietet. AWS PCS unterstützt den Slurm-Scheduler zur Verwaltung Ihrer Cluster-Workloads.

Topics

Slurm-Versionen in AWS STK.

SchedMD erweitert Slurm kontinuierlich mit neuen Funktionen, Optimierungen und Sicherheitspatches. SchedMD veröffentlicht in regelmäßigen Abständen eine neue Hauptversion und plant, bis zu 3 Versionen gleichzeitig zu unterstützen. AWS PCS ist so konzipiert, dass der Slurm-Controller automatisch mit Patch-Versionen aktualisiert wird.

Wenn SchedMD die Unterstützung für eine bestimmte Hauptversion einstellt, bezeichnet AWS PCS diese Version als End of Life (EOL). Nach EOL können mit dieser Version keine neuen Cluster mehr erstellt werden, obwohl bestehende Cluster bis zu 12 Monate lang ohne garantierten Support weiterlaufen können. AWS PCS sendet im Voraus eine Benachrichtigung, wenn eine Slurm-Hauptversion kurz vor EOL steht, damit Kunden wissen, wann sie ihre Cluster auf eine neuere unterstützte Version aktualisieren müssen.

Wir empfehlen Ihnen, für die Bereitstellung Ihres Clusters die neueste unterstützte Slurm-Version zu verwenden, um auf die neuesten Weiterentwicklungen und Verbesserungen zugreifen zu können.

Unterstützte Slurm-Versionen in AWS STK.

Die folgende Tabelle zeigt die unterstützten Slurm-Versionen sowie wichtige Daten und Informationen für jede Version.

Slurm-Version	Veröffentlichungsdatum von SchedMD	AWS Veröffentlichungsdatum von PCS	AWS PCS-EOL-Datum	Minimale kompatible AWS PCS-Agent enversion	Unterstützte AWS PCS-Beispiel-AMIs
25.11	11/6/2025	4/9/2026	5/31/2027	1.0.0-1	• `aws-pcs-sample_ami-al2023-x86_64-slurm-25.11` • `aws-pcs-sample_ami-al2023-arm64-slurm-25.11`
25,05	5/29/2025	10/16/2025	11/30/2026	1.0.0-1	• `aws-pcs-sample_ami-amzn2-x86_64-slurm-25.05`

Slurm-Version	Veröffentlichungsdatum von SchedMD	AWS Veröffentlichungsdatum von PCS	AWS PCS-EOL-Datum	Minimale kompatible AWS PCS-Agent enversion	Unterstützte AWS PCS-Beispiel-AMIs
					• `aws-pcs-sample_ami-amzn2-arm64-slurm-25.05`

Nicht unterstützte Slurm-Versionen in AWS STK.

Die folgende Tabelle zeigt Slurm-Versionen, die in AWS PCS nicht unterstützt werden.

Slurm-Version	Veröffentlichungsdatum von SchedMD	AWS Veröffentlichungsdatum von PCS	AWS PCS-EOL-Datum		
24.11	11/29/2024	5/14/2025	5/31/2026		
24,05	5/30/2024	12/18/2024	11/30/2025		
23,11	11/21/2023	8/28/2024	5/31/2025		

Versionshinweise für Slurm-Versionen in AWS STK.

Dieses Thema beschreibt wichtige Änderungen für jede Slurm-Version, die derzeit in AWS PCS unterstützt wird. Wir empfehlen Ihnen, die Änderungen zwischen der alten und der neuen Version zu überprüfen, wenn Sie Ihren Cluster aktualisieren.

Slurm 25.11

Die Änderungen wurden implementiert in AWS STK.

- Die Auditprotokolle von Scheduler werden jetzt getrennt nach `PCS_SCHEDULER_AUDIT_LOGS` Protokolltyp bereitgestellt. Dies vereinfacht die Problembehandlung und Prüfung, da die Protokollzustellung unabhängig gesteuert werden kann. Weitere Informationen finden Sie unter Scheduler-Prüfprotokolle in AWS PCS.
- Die beschleunigte Warteschlange ist standardmäßig aktiviert. Jobs, die aufgrund von Knotenproblemen fehlschlagen (z. B. Fehler bei unzureichender Kapazität), können mit der höchsten Planungspriorität in die Warteschlange gestellt werden. `sbatch --requeue=expedite` Dies wird durch die Einstellung `SchedulerParameters=enable_expedited_requeue` gesteuert.
- Der `requeue_delay` Parameter ist als benutzerdefinierte Clustereinstellung mit einer Standardeinstellung von 5 Sekunden verfügbar. Bisher war die Warteschlangenverzögerung an den Ablauf der Anmeldeinformationen gebunden (70 Sekunden). Administratoren können dies jetzt unabhängig über konfigurieren. `SchedulerParameters=requeue_delay=<seconds>`
- `HealthCheckNodeState`unterstützt jetzt den `START_ONLY` Wert, der das Health Check-Programm nur beim Start des Knotens ausführt (slurmd start).
- `CommunicationParameters=disable_http`pist standardmäßig so eingestellt, dass die in Slurm 25.11 eingeführten HTTP-Endpunkte (Metriken und Integritätstests) deaktiviert werden. Um diese Endpunkte wieder zu aktivieren, setzen Sie. `CommunicationParameters=enable_http` Weitere Informationen finden Sie unter Slurm-Metriken in AWS PCS.

Bekannte Probleme

- Slurm 25.11 validiert `AllowQOS` und `DenyQOS` partitioniert Einstellungen, auch wenn sie `AccountingStorageEnforce=QOS` nicht gesetzt sind. Wenn ein QOS, auf das in der Slurm-Buchhaltungsdatenbank verwiesen wird `AllowQOS` oder nicht `DenyQOS` existiert, wird der Vorgang mit einem schwerwiegenden Fehler `slurmctld` beendet. Stellen Sie sicher, dass alle in Partition `AllowQOS` und `DenyQOS` Einstellungen aufgelisteten QOS-Werte in der Accounting-Datenbank vorhanden sind, bevor Sie auf Slurm 25.11 aktualisieren oder diesen neu starten.
- Das `slurmd` Protokoll zeigt möglicherweise die Fehlermeldung an. `error: cannot create url_parser context for http_parser/libhttp_parser` Dies ist ein bekanntes Slurm-Problem, das auch dann auftritt, wenn `CommunicationParameters=disable_http` es gesetzt ist. Der Fehler kann problemlos ignoriert werden und hat keinen Einfluss auf den Clusterbetrieb.

Weitere Informationen zu Slurm 25.11 finden Sie in den folgenden Veröffentlichungen:

- Ankündigung der Veröffentlichung von SchedMD: https://www.schedmd.com/slurm-version-25-11-0-is-now-available/
- Versionshinweise zu SchedMD: https://github.com/SchedMD/slurm/blob/slurm-25.11/RELEASE_NOTES.md

Slurm 25.05

Die Änderungen wurden implementiert in AWS STK.

- Der Slurm requeue_on_resume_failure ist jetzt standardmäßig SchedulerParameter aktiviert.
- „stderr“ wurde als Option für entfernt, da es in Slurm 25.05 deaktiviert wurde. LogTimeFormat
- AWS PCS unterstützt die Multi-cluster Sackd-Konfiguration: Der Anmeldeknoten kann auf mehrere Cluster zugreifen.

Weitere Informationen zu Slurm 25.05 finden Sie in den folgenden Publikationen:

- Ankündigung der Veröffentlichung von SchedMD: https://www.schedmd.com/slurm-version-25-05-0-is-now-available/
- Versionshinweise zu SchedMD: https://github.com/SchedMD/slurm/blob/slurm-25-05-0-1/RELEASE_NOTES.md

Slurm 24.11

Die Änderungen wurden implementiert in AWS STK.

- AWS PCS unterstützt Slurm Accounting. Weitere Informationen finden Sie unter Slurm-Buchhaltung in AWS STK..

Weitere Informationen zu Slurm 24.11 finden Sie in den folgenden Veröffentlichungen:

- Ankündigung der Veröffentlichung von SchedMD
- Versionshinweise zu SchedMD

Slurm 24.05

Die Änderungen wurden implementiert in AWS STK.

- Das neue Slurm Step Manager-Modul ist jetzt standardmäßig in AWS PCS aktiviert. Dieses Modul bietet erhebliche Vorteile, da das Schrittmanagement vom zentralen Controller auf die Rechenknoten verlagert wird, wodurch die Parallelität der Systeme in Umgebungen mit starker Schrittnutzung erheblich verbessert wird. Um diese Konfiguration zu unterstützen und die Ausführung besser zu isolieren `Prolog` und zu `Epilog` verarbeiten, wurden neue Prolog-Flags (`Contain`,`Alloc`) aktiviert.
- Die hierarchische Kommunikation vom Controller zu den Rechenknoten wird aktiviert, um die Kommunikation zwischen Slurm-Knoten zu optimieren und so die Skalierbarkeit und Leistung zu verbessern. Darüber hinaus verwendet die Routing-Konfiguration jetzt Partitionsknotenlisten für die Kommunikation vom Controller anstelle des Standard-Routing-Algorithmus des Plugins, wodurch die Systemstabilität verbessert wird.
- Ein neues Hash-Plugin `HashPlugin=hash/sha3` ersetzt das vorherige`hash/k12 plugin`. Dies ist jetzt standardmäßig in AWS PCS-Clustern aktiviert.
- Die Slurm-Controller-Logs enthalten jetzt erweiterte Auditing-Funktionen für alle eingehenden Remote Procedure Calls (RPC). `slurmctld` Die Protokolle enthalten die Quelladresse, den authentifizierten Benutzer und den RPC-Typ vor der Verbindungsverarbeitung.

Weitere Informationen zu Slurm 24.05 finden Sie in den folgenden Veröffentlichungen:

- Ankündigung der Veröffentlichung von SchedMD
- Versionshinweise zu SchedMD

Slurm 23.11

Slurm-Einstellungen, die du ändern kannst AWS STK.

- Die `SuspendTime` Standardeinstellung ist. `60` Verwenden Sie den AWS `scaleDownIdleTimeInSeconds` PCS-Konfigurationsparameter, um ihn festzulegen. Weitere Informationen finden Sie unter dem `scaleDownIdleTimeInSeconds`Parameter des `ClusterSlurmConfiguration` Datentyps in der AWS PCS-API-Referenz.
- Der `MaxJobCount` Wert und `MaxArraySize` basiert auf der Größe, die Sie für den Cluster auswählen. Weitere Informationen finden Sie unter dem `size`Parameter der `CreateCluster` API-Aktion in der AWS PCS-API-Referenz.

- Die `SelectTypeParameters` Slurm-Einstellung ist standardmäßig auf. `CR_CPU` Sie können ihn als Wert angeben, `slurmCustomSettings` um ihn bei der Erstellung eines Clusters festzulegen. Weitere Informationen finden Sie im `slurmCustomSettings`Parameter der `CreateCluster` API-Aktion und SlurmCustomSettingin der AWS PCS-API-Referenz.
- Sie können `Prolog` und `Epilog` auf Clusterebene festlegen. Sie können es als Wert angeben`slurmCustomSettings`, um es festzulegen, wenn Sie einen Cluster erstellen. Weitere Informationen finden Sie unter `CreateCluster`und SlurmCustomSettingin der AWS PCS-API-Referenz.
- Sie können `Weight` und `RealMemory` auf der Ebene der Compute-Knotengruppen festlegen. Sie können es als Wert angeben, `slurmCustomSettings` um es festzulegen, wenn Sie eine Compute-Knotengruppe erstellen. Weitere Informationen finden Sie unter `CreateComputeNodeGroup`und SlurmCustomSettingin der AWS PCS-API-Referenz.

Häufig gestellte Fragen zu Slurm-Versionen in AWS STK.

AWS PCS unterstützt weiterhin mehrere Slurm-Versionen. Wenn eine neue Slurm-Version eingeführt wird, bietet AWS PCS technischen Support und Sicherheitspatches, bis diese Version das Ende des Supports (EOS) von SchedMD erreicht. AWS PCS bezeichnet das EOS-Datum für eine Slurm-Version aus terminologischen Gründen als End of Life (EOL). AWS

Wie lang dauert AWS PCS unterstützt eine Slurm-Version?

AWS Die PCS-Unterstützung für Slurm-Versionen entspricht den Supportzyklen von SchedMD für Hauptversionen. AWS PCS unterstützt die aktuelle Version und die beiden neuesten vorherigen Hauptversionen. Wenn SchedMD eine neue Hauptversion veröffentlicht, beendet AWS PCS die Unterstützung für die älteste unterstützte Version. AWS PCS veröffentlicht neue Hauptversionen von Slurm so schnell wie möglich, aber es kann zu Verzögerungen zwischen der Veröffentlichung von SchedMD und ihrer Verfügbarkeit in PCS kommen. AWS

Wie erhalten meine Cluster neue Slurm-Patch-Versionen?

Um Fehler zu beheben und Sicherheitsfixes zu beheben, ist AWS PCS so konzipiert, dass Patches automatisch auf Cluster-Controllern installiert werden, die unter internen Dienstkonten ausgeführt werden. Um Patches auf EC2-Instances in Ihrem zu installieren AWS-Konto, aktualisieren Sie das Amazon Machine Image (AMI) für Ihre Compute-Knotengruppen und aktualisieren Sie die Compute-Knotengruppen, um das aktualisierte AMI zu verwenden. Weitere Informationen finden Sie unter Benutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS.

> **Note**
>
> Slurm-Controller sind nicht verfügbar, solange wir sie aktualisieren. Laufende Jobs sind nicht betroffen. Jobs, die eingereicht wurden, bevor der Controller des Clusters nicht mehr verfügbar war, werden zurückgehalten, bis der Controller verfügbar ist.

Wie werde ich über ein bevorstehendes EOL-Event für die Slurm-Version informiert?

Wir senden Ihnen 6 Monate vor dem EOL-Datum eine E-Mail-Nachricht. Wir senden Ihnen jeden Monat vor dem EOL-Datum eine E-Mail-Nachricht mit einer letzten E-Mail-Nachricht 1 Woche vor dem EOL-Datum. Nach dem EOL-Datum senden wir 12 Monate lang monatliche E-Mail-Nachrichten an Kunden, die AWS PCS-Cluster mit EOL-Slurm-Versionen betreiben. Wir können einen Cluster mit einer EOL-Slurm-Version aussetzen, wenn für diese Version Sicherheitslücken festgestellt werden.

Wie kann ich feststellen, ob auf der von meinem Cluster verwendeten Slurm-Version eine EOL-Slurm-Version ausgeführt wird?

Wir senden Ihnen eine E-Mail-Nachricht, um Sie darüber zu informieren, dass Sie einen laufenden Cluster mit einer EOL-Slurm-Version haben. Wir senden eine Warnung zu den AWS Health Dashboard Alerts, die die Details Ihrer Cluster mit EOL-Slurm-Versionen enthält. Sie können auch die AWS PCS-Konsole verwenden, um die Cluster mit EOL-Slurm-Versionen zu identifizieren.

Was muss ich tun, wenn meine Slurm-Version kurz vor oder nach EOL liegt?

Erstellen Sie einen neuen Cluster mit einer neueren unterstützten Version von Slurm und aktualisieren Sie die Slurm-Version in Ihren Compute-Knotengruppen-AMIs. Die Slurm-Version in Ihren AMIs und laufenden EC2-Instances darf nicht mehr als 2 Versionen hinter der Slurm-Version des Clusters liegen. Weitere Informationen finden Sie unter Benutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS.

Was passiert, wenn ich bis zum EOL-Datum nicht zu einer neueren Version von Slurm wechsle?

Mit einer EOL-Slurm-Version können Sie keine neuen Cluster erstellen. Bestehende Cluster können bis zu 12 Monate ohne AWS Support betrieben werden, und es sind keine sofortigen Maßnahmen erforderlich, um ihren Betrieb aufrechtzuerhalten. Nach dem EOL-Datum können Support, Sicherheitsupdates und Verfügbarkeit nicht garantiert werden. Wir können einen Cluster aus Sicherheitsgründen aussetzen. Wir empfehlen Ihnen dringend, eine unterstützte Slurm-Version zu verwenden, um die Sicherheit und den Support für Ihre AWS PCS-Cluster zu gewährleisten.

Was sind die Risiken beim Betrieb eines Clusters mit EOL-Slurm-Versionen?

Cluster mit EOL-Slurm-Versionen bergen erhebliche Sicherheits- und Betriebsrisiken. Ohne die aktive Überwachung durch SchedMD könnten Sicherheitslücken unentdeckt bleiben oder nicht behoben werden. Wenn kritische Sicherheitslücken entdeckt werden, können wir Ihre Cluster sofort sperren.

Was passiert mit meinen Jobs, Cluster-Rechen-, Speicher- und Netzwerkressourcen, wenn mein Cluster gesperrt wird?

Alle von AWS PCS verwalteten Ressourcen werden beendet. Dazu gehören der Slurm-Controller, Rechenknotengruppen und EC2-Instances. Alle Jobs, die auf Compute-Instances ausgeführt werden, werden sofort beendet und der Cluster wechselt in einen angehaltenen Zustand. Customer-managedRessourcen, wie z. B. externe Dateisysteme, bleiben intakt. Sie können die AWS PCS-Konsole und API-Aktionen verwenden, um auf die Konfiguration des Clusters zuzugreifen.

Kann ich einen angehaltenen Cluster neu starten, um die verbleibenden Jobs wieder aufzunehmen?

Nein, Sie können einen unterbrochenen Cluster nicht neu starten. Sie können die Konfiguration Ihres suspendierten Clusters verwenden, um einen neuen Cluster mit einer unterstützten Slurm-Version zu erstellen. Sie können Ihre verbleibenden Jobs ausführen, wenn Sie sie in einem externen Dateisystem gespeichert haben.

Kann ich eine Verlängerung über die 12-monatige Nachfrist hinaus beantragen?

Nein, Sie können keine Verlängerung für den Betrieb Ihres Clusters nach Ablauf der 12-monatigen Kulanzzeit beantragen. Wir stellen Ihnen die verlängerte Frist zur Verfügung, um Ihnen bei der Umstellung auf eine unterstützte Slurm-Version zu helfen. Um Störungen Ihres Clusterbetriebs zu vermeiden, empfehlen wir Ihnen, zu wechseln, bevor Ihre Slurm-Version EOL erreicht.

Slurm-Buchhaltung in AWS STK.

> **Note**
>
> Die Buchhaltung wird für Slurm 24.11 oder höher unterstützt.

Sie können die Kontoführung auf Ihren neuen AWS PCS-Clustern aktivieren, um die Clusternutzung zu überwachen, Ressourcenlimits durchzusetzen und eine detaillierte Zugriffskontrolle für bestimmte

Warteschlangen oder Rechenknotengruppen zu verwalten. AWS PCS erstellt und verwaltet die Accounting-Datenbank für Ihren Cluster, sodass Sie keine eigene separate Accounting-Datenbank erstellen und verwalten müssen. AWS PCS verwendet die Buchhaltungsfunktion in Slurm. Weitere Informationen zur Buchhaltungsfunktion in Slurm finden Sie in der Slurm-Dokumentation auf SchedMD.

Um Accounting zu verwenden, aktivieren Sie es, wenn Sie einen neuen Cluster erstellen, und legen Sie optional Accounting-Parameter fest. Wenn Ihr Clusterstatus lautet `Active` und über Compute-Knotengruppen verfügt, können Sie eine Verbindung zur Linux-Shell eines Anmeldeknotens herstellen, um Abrechnungsfunktionen auszuführen, z. B. das Anzeigen von Auftragsdaten mit dem `sacct` Slurm-Befehl.

AWS PCS console

Aktivieren Sie auf der Seite Cluster erstellen unter Scheduler-Einstellungen die Option Accounting.

AWS PCS API

Geben Sie die `accounting` Konfiguration in Ihrem Aufruf der `CreateCluster` API-Aktion an. Stellen Sie im `accounting` Objekt den Wert `mode` auf ein`STANDARD`. Weitere Informationen finden Sie unter CreateClusterAccounting in der AWS PCS API-Referenz.

Im folgenden Beispiel wird die Aktion AWS CLI zum Aufrufen der `CreateCluster` API-Aktion verwendet. Der Parameterwert substring `accounting='{mode=STANDARD}'` ermöglicht die Abrechnung.

```
aws pcs create-cluster --cluster-name cluster-name \
                       --scheduler type=SLURM,version=25.11 \
                       --size SMALL \
                       --networking subnetIds=cluster-subnet-
id,securityGroupIds=cluster-security-group-id \
                       --slurm-configuration
 scaleDownIdleTimeInSeconds=180,accounting='{mode=STANDARD}',slurmCustomSettings='[{paramete
```

Important

Wenn Sie die Buchhaltung aktivieren, fallen zusätzliche Abrechnungsgebühren an. Weitere Informationen finden Sie auf der AWS PCS-Preisseite.

Kontoführungseinstellungen ändern

Sie können die Kontoführung auf vorhandenen Clustern aktivieren oder deaktivieren, ohne Ihre Infrastruktur neu aufbauen zu müssen. Weitere Informationen finden Sie unter Aktualisierung eines Clusters in AWS PCS.

Wenn Sie die Kontoführung deaktivieren, wird die Abrechnung für die Abrechnungsfunktion beendet, sobald der Cluster den `UPDATING` Status erreicht. Wenn Sie die Kontoführung aktivieren, beginnt die Abrechnung, wenn der Cluster erfolgreich in den `ACTIVE` Status zurückkehrt.

Schlüsselkonzepte für die Slurm-Buchhaltung in AWS STK.

Die folgenden Konzepte sind spezifisch für AWS PCS und steuern, wie AWS PCS die Slurm-Buchhaltung implementiert.

Buchhaltungsdatenbank

AWS PCS speichert Ihre Buchhaltungsdaten in einer Datenbank AWS-Konto , die in einer eigenen Datenbank erstellt AWS wurde. Sie haben keinen Zugriff auf `slurmdbd.conf`.

Standardlöschzeit

Diese AWS PCS-Einstellung gibt den Aufbewahrungszeitraum (in Tagen) für alle Arten von Buchhaltungsdatensätzen an (Aufträge, Ereignisse, Reservierungen, Schritte, Aussetzungen, Transaktionen, Nutzungsdaten). Wenn der Wert beispielsweise 30 ist, bewahrt AWS PCS Buchhaltungsdaten 30 Tage lang auf. Sie geben diesen Wert an, wenn Sie den Cluster erstellen. Wenn Sie keinen Wert angeben, speichert AWS PCS die Buchhaltungsdaten auf unbestimmte Zeit in der Datenbank.

AWS PCS console

Sie geben die Standardlöschzeit als Teil der Schritte zum Erstellen eines Clusters an. Aktivieren Sie auf der Seite Cluster erstellen die Kontoführung. Geben Sie unter Scheduler-Einstellungen einen ganzzahligen Wert für die Standard-Löschzeit (Tage) ein.

AWS PCS API

Geben Sie den `defaultPurgeTimeInDays` als Teil der `accounting` Informationen an, die Sie in Ihrem Aufruf der `CreateCluster` API-Aktion angeben. Weitere Informationen finden Sie unter CreateClusterAccounting in der AWS PCS-API-Referenz.

> **Note**
>
> Wenn Sie die AWS PCS-API verwenden, um einen Cluster zu erstellen, `defaultPurgeTimeInDays` ist der Standardwert für ein gültiger Wert `-1` und `0` kein gültiger Wert.

Durchsetzung der Rechnungslegungsrichtlinien

Diese Einstellung bestimmt, wie strikt Slurm die Regeln für die Einreichung von Jobs, Ressourcenlimits und Abrechnungsrichtlinien für Ihren Cluster durchsetzt. Diese Einstellung entspricht dem `AccountingStorageEnforce` Parameter in der Datei Ihres Clusters`slurm.conf`. Sie können eine beliebige Kombination von Durchsetzungsoptionen auswählen. Wenn Sie keine Optionen auswählen, gelten für Jobs im Cluster keine buchhalterischen Einschränkungen. AWS PCS unterstützt die folgenden Optionen:

- Verbände — Zuordnung von Aufträgen zu Konten
- Grenzen — Ressourcenknappheit
- QoS — Anforderungen an die Servicequalität
- abgesicherter Modus — garantierte Fertigstellung innerhalb bestimmter Grenzen
- nosteps — Deaktiviert die Schrittabrechnung
- nojobs — deaktiviert die Auftragsabrechnung

Weitere Informationen zu diesen Optionen finden Sie in der Slurm-Dokumentation auf SchedMD.

AWS PCS console

Sie legen die Optionen im Rahmen der Schritte zum Erstellen eines Clusters fest. Aktivieren Sie auf der Seite Cluster erstellen die Kontoführung. Wählen Sie die gewünschten Optionen aus der Dropdownliste Durchsetzung von Accounting-Richtlinien unter Scheduler-Einstellungen aus.

AWS PCS API

In Slurm werden diese Optionen in der Datei eines Clusters festgelegt. `slurm.conf` Sie haben keinen direkten Zugriff auf den `slurm.conf` für Ihren AWS PCS-Cluster. Stattdessen stellen Sie die `CreateCluster` API-Aktion `SlurmCustomSettings` bereit, wenn Sie einen Cluster erstellen. Weitere Informationen finden Sie CreateClusterin der AWS PCS-API-Referenz.

Rufen Sie die Accounting-Konfiguration für ein vorhandenes ab AWS PCS-Cluster

Die Slurm-Accounting-Konfiguration ist in der Slurm-Konfiguration für Ihren Cluster enthalten.

AWS PCS console

1. Wählen Sie im Navigationsbereich Cluster aus.
2. Wählen Sie den Clusternamen aus der Liste aus.
3. Suchen Sie auf der Registerkarte Konfiguration unter Slurm-Konfiguration nach der Accounting-Konfiguration

AWS PCS API

Verwenden Sie die `GetCluster` API-Aktion, um die Cluster-Konfiguration abzurufen. Die Accounting-Konfiguration finden Sie in der`slurmConfiguration`. Die Einstellung für `mode` und der Wert von `defaultPurgeTimeInDays` liegen unter`accounting`. Die ausgewählten Optionen zur Durchsetzung der Rechnungslegungsrichtlinien befinden sich unter`slurmCustomSettings`. Weitere Informationen finden Sie GetClusterin der AWS PCS-API-Referenz.

Schlurm-REST-API ein AWS STK.

AWS PCS bietet verwalteten Support für die native REST-API von Slurm und stellt eine HTTP-Schnittstelle für die `slurmrestd` programmatische Cluster-Interaktion bereit. Sie können Jobs einreichen, den Clusterstatus überwachen und Ressourcen über Standard-HTTP-Anfragen verwalten, ohne direkten Shell-Zugriff auf Ihren Cluster zu benötigen.

Häufige Anwendungsfälle

Die Slurm-REST-API unterstützt verschiedene Integrationsszenarien:

- Integration von Webanwendungen: Erstellen Sie benutzerdefinierte Frontends und Webanwendungen, die Jobs direkt einreichen und verwalten.
- Jupyter Notebook-Integration: Ermöglicht Forschern, Jobs von Notebook-Umgebungen aus einzureichen, ohne ihren Entwicklungsworkflow zu verlassen.

- Integration von Partnerlösungen: Connect HPC-Tools und Workflow-Manager von Drittanbietern mit Ihren AWS PCS-Clustern.
- Programmatisches Clustermanagement: Automatisieren Sie die Workflows für die Auftragsübergabe, Überwachung und Ressourcenverwaltung.
- Workflows für die Datenverarbeitung in der Forschung: Support Sie akademische und betriebliche Forschungsumgebungen, in denen API-driven Auftragsmanagement erforderlich ist.

Anforderungen und Einschränkungen

Bevor Sie die Slurm-REST-API verwenden, überprüfen Sie diese Details:

- Ihr Cluster muss Slurm Version 25.05 oder höher verwenden.
- Auf den API-Endpunkt kann nur über eine private IP-Adresse innerhalb der VPC Ihres Clusters zugegriffen werden.
- Ihre Cluster-Sicherheitsgruppe muss HTTP-Verkehr auf Port 6820 zulassen.
- Für die Authentifizierung sind JWT-Token mit bestimmten Benutzeridentitätsansprüchen erforderlich.

Zu den aktuellen Einschränkungen gehören:

- Von generierte Token `scontrol token` werden nicht unterstützt.
- `X-SLURM-USER-NAME`Die Imitation von Headern ist nicht verfügbar.
- Für einige Funktionen muss die Slurm-Accounting-Funktion aktiviert sein.
- Nicht kompatibel mit dem Slurm-CLI-Filter-Plugin-Mechanismus.
- Verbindungen zum REST-API-Endpunkt sind nicht mit TLS verschlüsselt.

Themen

Aktivierung der Slurm-REST-API in AWS STK.

Aktivieren Sie die Slurm-REST-API für den Zugriff auf die HTTP-Schnittstelle Ihres Clusters für die programmatische Auftragsverwaltung und -überwachung. Sie können diese Funktion während der Clustererstellung aktivieren oder einen vorhandenen Cluster aktualisieren, der die Anforderungen erfüllt.

Voraussetzungen

Bevor Sie die Slurm-REST-API aktivieren, stellen Sie sicher, dass Sie über Folgendes verfügen:

- Cluster-Version: Slurm-Version 25.05 oder höher.
- Sicherheitsgruppe: Regeln, die HTTP-Verkehr auf Port 6820 von Ihren gewünschten Quellen zulassen.

Verfahren

Um die Slurm-REST-API auf einem neuen Cluster zu aktivieren

AWS-Managementkonsole

1. Öffnen Sie die AWS PCS-Konsole unter. https://console.aws.amazon.com/pcs/
2. Wählen Sie Cluster erstellen.
3. Wählen Sie unter Cluster-Details die Slurm-Version 25.05 oder höher aus.
4. Konfigurieren Sie die anderen Cluster-Einstellungen nach Bedarf.
5. Stellen Sie im Abschnitt Scheduler-Konfiguration die REST-API auf Aktiviert ein.
6. Konfigurieren Sie Ihre Cluster-Sicherheitsgruppe so, dass HTTP-Verkehr auf Port 6820 von Ihren gewünschten Quellen zugelassen wird.
7. Schließen Sie den Vorgang zur Clustererstellung ab.

AWS CLI

1. Fügen Sie bei der Erstellung Ihres Clusters eine Slurm-REST-Konfiguration hinzu.

```
aws pcs create-cluster --region region \
    --cluster-name my-cluster \
```

```
    --scheduler type=SLURM, version=25.05 \
    --size SMALL \
    --networking subnetIds=subnet-ExampleId1,securityGroupIds=sg-ExampleId1 \
    --slurm-configuration slurmRest='{mode=STANDARD}'
```

2. Konfigurieren Sie Ihre Cluster-Sicherheitsgruppe so, dass HTTP-Verkehr auf Port 6820 von Ihren gewünschten Quellen aus zugelassen wird.

Um die Slurm-REST-API auf einem vorhandenen Cluster zu aktivieren

AWS-Managementkonsole

1. Öffnen Sie die AWS PCS-Konsole unter. https://console.aws.amazon.com/pcs/
2. Wählen Sie Ihren Cluster aus der Liste aus.
3. Stellen Sie in den Cluster-Details sicher, dass Ihr Cluster Slurm Version 25.05 oder höher verwendet.
4. Wählen Sie Cluster bearbeiten.
5. Stellen Sie im Abschnitt Scheduler-Konfiguration die REST-API auf Aktiviert ein.
6. Wählen Sie Cluster aktualisieren, um die Änderungen zu übernehmen.
7. Konfigurieren Sie Ihre Cluster-Sicherheitsgruppe so, dass HTTP-Verkehr auf Port 6820 von Ihren gewünschten Quellen zugelassen wird.

AWS CLI

1. Aktualisieren Sie Ihren Cluster mit einer Slurm-REST-Konfiguration, wie in diesem Beispiel.

```
aws pcs update-cluster --cluster-identifier my-cluster \
    --slurm-configuration 'slurmRest={mode=STANDARD}'
```

2. Konfigurieren Sie Ihre Cluster-Sicherheitsgruppe so, dass HTTP-Verkehr auf Port 6820 aus Ihren gewünschten Quellen zugelassen wird.

Was passiert nach der Aktivierung

Wenn Sie die REST-API aktivieren, führt AWS PCS automatisch:

- Generiert einen JWT-Signaturschlüssel und speichert ihn in AWS Secrets Manager.

- Macht den API-Endpunkt `https://<clusterPrivateIpAddress>:6820` in Ihrer VPC verfügbar.
- Aktualisiert Ihre Cluster-Konfiguration, sodass die REST-API-Endpunktdetails angezeigt werden.

Sie können sich jetzt authentifizieren und die REST-API für die Auftragsverwaltung und Clustervorgänge verwenden.

Authentifizierung mit der Slurm-REST-API in AWS STK.

Die Slurm-REST-API in AWS PCS verwendet die JSON-Web-Token-Authentifizierung (JWT), um einen sicheren Zugriff auf Ihre Cluster-Ressourcen zu gewährleisten. AWS PCS stellt einen verwalteten Signaturschlüssel bereit, der in AWS Secrets Manager gespeichert ist und den Sie verwenden, um JWT-Token zu generieren, die bestimmte Benutzeridentitätsansprüche enthalten.

Voraussetzungen

Bevor Sie sich mit der Slurm-REST-API authentifizieren, stellen Sie sicher, dass Sie über Folgendes verfügen:

- Cluster-Konfiguration: AWS PCS-Cluster mit Slurm 25.05+ und aktivierter REST-API.
- AWS-Berechtigungen: Zugriff auf AWS Secrets Manager für den JWT-Signaturschlüssel.
- Benutzerinformationen: Benutzername, POSIX-Benutzer-ID und eine oder mehrere POSIX-Gruppen-IDs für Ihr Cluster-Konto.
- Netzwerkzugriff: Konnektivität innerhalb der VPC Ihres Clusters mit einer Sicherheitsgruppe, die Port 6820 zulässt.

Verfahren

Um die Adresse des Slurm-REST-API-Endpunkts abzurufen

AWS-Managementkonsole

1. Öffnen Sie die AWS PCS-Konsole unter. https://console.aws.amazon.com/pcs/
2. Wählen Sie Ihren Cluster aus der Liste aus.
3. Suchen Sie in den Cluster-Konfigurationsdetails den Abschnitt Endpoints.
4. Notieren Sie sich die private IP-Adresse und den Port für die Slurm-REST-API (slurmrestd).

5. Sie können API-Aufrufe tätigen, indem Sie ordnungsgemäß formatierte HTTP-Anfragen an diese Adresse senden.

AWS CLI

1. Fragen Sie Ihren Clusterstatus mit `aws pcs get-cluster` ab. Suchen Sie in dem `endpoints` Feld in der Antwort nach dem `SLURMRESTD` Endpunkt. Ein Beispiel:

```
"endpoints": [
      {
          "type": "SLURMCTLD",
          "privateIpAddress": "192.0.2.1",
          "port": "6817"
      },
      {
          "type": "SLURMRESTD",
          "privateIpAddress": "192.0.2.1",
          "port": "6820"
      }
  ]
```

2. Sie können API-Aufrufe tätigen, indem Sie ordnungsgemäß formatierte HTTP-Anfragen an senden `http://<privateIpAddress>:<port>/`

Um den JWT-Signaturschlüssel abzurufen

1. Öffnen Sie die AWS PCS-Konsole unter. https://console.aws.amazon.com/pcs/
2. Wählen Sie Ihren Cluster aus der Liste aus.
3. Suchen Sie in den Cluster-Konfigurationsdetails den Abschnitt Scheduler-Authentifizierung.
4. Notieren Sie sich den ARN und die Version des JSON-Web-Token-Schlüssels (JWT).
5. Verwenden Sie den AWS CLI , um den Signaturschlüssel von Secrets Manager abzurufen:

```
aws secretsmanager get-secret-value --secret-
id arn:aws:secretsmanager:region:account:secret:name --version-id version
```

Um ein JWT-Token zu generieren

1. Erstellen Sie ein JWT mit den folgenden erforderlichen Ansprüchen:

- `exp`— Ablaufzeit in Sekunden seit 1970 für das JWT
- `iat`— Aktuelle Zeit in Sekunden seit 1970
- `sun`— Der Benutzername für die Authentifizierung
- `uid`— Die POSIX-Benutzer-ID
- `gid`— Die POSIX-Gruppen-ID
- `id`— Zusätzliche POSIX-Identitätseigenschaften
  - `gecos`— Feld für Benutzerkommentare, das häufig verwendet wird, um einen für Menschen lesbaren Namen zu speichern
  - `dir`— Heimatverzeichnis des Benutzers
  - `shell`— Standard-Shell des Benutzers
  - `gids`— Liste zusätzlicher POSIX-Gruppen-IDs, in denen sich der Benutzer befindet

2. Signieren Sie das JWT mit dem Signaturschlüssel, den Sie von Secrets Manager abgerufen haben.
3. Legen Sie eine angemessene Ablaufzeit für das Token fest.

> **Note**
>
> Als Alternative zum `sun` Antrag können Sie eine der folgenden Angaben machen:
>
> - `username`
> - Ein benutzerdefinierter Feldname, den Sie über den `userclaimfield` in der `AuthAltParameters Slurm custom settings`
> - Ein `name` Feld innerhalb des `id` Antrags

Um API-Anfragen zu authentifizieren

1. Fügen Sie das JWT-Token mithilfe einer der folgenden Methoden in Ihre HTTP-Anfragen ein:
   - Trägertoken — Header hinzufügen `Authorization: Bearer <jwt>`
   - Slurm-Header — Header hinzufügen `X-SLURM-USER-TOKEN: <jwt>`
2. Stellen Sie HTTP-Anfragen an den REST-API-Endpunkt:

Hier ist ein Beispiel für den Zugriff auf die `/ping` API mithilfe von Curl und dem `Authorized: Bearer` Header.

```
curl -X GET -H "Authorization: Bearer <jwt>" \
      http://<privateIpAddress>:6820/slurm/v0.0.43/ping
```

Beispiel für eine JWT-Generierung

Rufen Sie den JWT-Signaturschlüssel des AWS PCS-Clusters ab und speichern Sie ihn als lokale Datei. Ersetzen Sie Werte für aws-region, secret-arn und secret version durch Werte, die für Ihren Cluster geeignet sind.

```
#!/bin/bash
SECRET_KEY=$(aws secretsmanager get-secret-value \
  --region aws-region \
  --secret-id secret-arn \
  --version-stage secret-version \
  --query 'SecretString' \
  --output text)
echo "$SECRET_KEY" | base64 --decode > jwt.key
```

Dieses Python-Beispiel zeigt, wie der Signaturschlüssel verwendet wird, um ein JWT-Token zu generieren:

```
#!/usr/bin/env python3

import sys
import os
import pprint
import json
import time
from datetime import datetime, timedelta, timezone
from jwt import JWT
from jwt.jwa import HS256
from jwt.jwk import jwk_from_dict
from jwt.utils import b64decode,b64encode
if len(sys.argv) != 3:
    sys.exit("Usage: gen_jwt.py [jwt_key_file] [expiration_time_seconds]")
SIGNING_KEY = sys.argv[1]
EXPIRATION_TIME = int(sys.argv[2])
```

```
with open(SIGNING_KEY, "rb") as f:
    priv_key = f.read()
signing_key = jwk_from_dict({
    'kty': 'oct',
    'k': b64encode(priv_key)
})
message = {
    "exp": int(time.time() + EXPIRATION_TIME),
    "iat": int(time.time()),
    "sun": "ec2-user",
    "uid": 1000,
    "gid": 1000,
    "id": {
        "gecos": "EC2 User",
        "dir": "/home/ec2-user",
        "gids": [1000],
        "shell": "/bin/bash"
    }
}
a = JWT()
compact_jws = a.encode(message, signing_key, alg='HS256')
print(compact_jws)
```

Das Skript druckt ein JWT auf den Bildschirm.

```
abcdefgtjwttoken...
```

Verwendung der Slurm-REST-API für die Auftragsverwaltung in AWS STK.

Überblick über die Slurm-REST-API

Die Slurm-REST-API bietet programmatischen Zugriff auf Cluster-Management-Funktionen über HTTP-Anfragen. Wenn Sie diese Hauptmerkmale verstehen, können Sie die API mit AWS PCS effektiv nutzen:

- Zugriffsprotokoll: Die API verwendet HTTP (nicht HTTPS) für die Kommunikation innerhalb des privaten Netzwerks Ihres Clusters.
- Verbindungsdetails: Greifen Sie über die private IP-Adresse Ihres Clusters und den `slurmrestd` Port (normalerweise 6820) auf die API zu. Das vollständige Basis-URL-Format ist`http://`*`<privateIpAddress>`*`:6820`.

- API-Versionierung: Die API-Version entspricht Ihrer Slurm-Installation. Verwenden Sie für Slurm 25.05 Version v0.0.43. Die Versionsnummer ändert sich mit jeder Slurm-Version. Die derzeit unterstützten API-Versionen finden Sie in den Slurm-Versionshinweisen.
- URL-Struktur: Die URL-Struktur für die Slurm-REST-API lautet. `http://<privateIpAddress>:<port>/<api-version>/<endpoint>` Detaillierte Nutzungsinformationen für REST-API-Endpunkte finden Sie in der Slurm-Dokumentation.

Spezifische Informationen zur Arbeit mit der Slurm-REST-API finden Sie in der Slurm-Dokumentation des REST-Clients.

Voraussetzungen

Bevor Sie die Slurm-REST-API verwenden, stellen Sie sicher, dass Sie über Folgendes verfügen:

- Cluster-Konfiguration: AWS PCS-Cluster mit Slurm 25.05+ und aktivierter REST-API.
- Authentifizierung: Gültiges JWT-Token mit korrekten Angaben zur Benutzeridentität.
- Netzwerkzugriff: Konnektivität innerhalb der VPC Ihres Clusters mit einer Sicherheitsgruppe, die Port 6820 zulässt.

Verfahren

Um einen Job mit der REST-API einzureichen

1. Erstellen Sie eine Anfrage zur Einreichung eines Jobs mit den erforderlichen Parametern:

```
{
  "job": {
    "name": "my-job",
    "partition": "compute",
    "nodes": 1,
    "tasks": 1,
    "script": "#!/bin/bash\necho 'Hello from Slurm REST API'",
    "environment": ["PATH=/usr/local/bin:/usr/bin:/bin"]
  }
}
```

2. Reichen Sie den Job mit einer HTTP-POST-Anfrage ein:

```
curl -X POST \
```

```
  -H "Authorization: Bearer <jwt>" \
  -H "Content-Type: application/json" \
  -d '<job-json>' \
  https://<privateIpAddress>:6820/slurm/v0.0.43/job/submit
```

3. Notieren Sie sich die in der Antwort zurückgegebene Job-ID zu Überwachungszwecken.

Um den Jobstatus zu überwachen

1. Informationen zu einem bestimmten Job abrufen:

```
curl -X GET -H "Authorization: Bearer <jwt>" \
    https://<privateIpAddress>:6820/slurm/v0.0.43/job/<job-id>
```

2. Listet alle Jobs für den authentifizierten Benutzer auf:

```
curl -X GET -H "Authorization: Bearer <jwt>" \
    https://<privateIpAddress>:6820/slurm/v0.0.43/jobs
```

So brechen Sie einen Auftrag ab

- Senden Sie eine DELETE-Anfrage, um einen bestimmten Job zu stornieren:

```
curl -X DELETE -H "Authorization: Bearer <jwt>" \
    https://<privateIpAddress>:6820/slurm/v0.0.43/job/<job-id>
```

Häufig gestellte Fragen zur Slurm-REST-API in AWS STK.

In diesem Abschnitt werden häufig gestellte Fragen zur Slurm-REST-API in AWS PCS beantwortet.

Was ist die Slurm-REST-API?

Die Slurm-REST-API ist eine HTTP-Schnittstelle, die es Ihnen ermöglicht, programmgesteuert mit dem Slurm-Workload-Manager zu interagieren. Sie können Standard-HTTP-Methoden wie GET, POST und DELETE verwenden, um Jobs zu senden, den Clusterstatus zu überwachen und Ressourcen zu verwalten, ohne dass Sie über die Befehlszeile auf den Cluster zugreifen müssen.

Kann ich Tokens verwenden, die von generiert wurden? **`scontrol token`**

Nein, die `scontrol token` Standardausgabe ist nicht mit AWS PCS kompatibel. Die PCS Slurm REST-API benötigt erweiterte JWT-Token, die spezifische Identitätsansprüche enthalten, darunter username (`sun`), POSIX-Benutzer-ID (`uid`) und Gruppen-IDs (). `gids` Bei Standard-Slurm-Token fehlen diese erforderlichen Ansprüche und sie werden von der API abgelehnt.

Kann ich von außerhalb meiner VPC auf die API zugreifen?

Nein, auf den REST-API-Endpunkt kann nur von Ihrer VPC aus über die private IP-Adresse des Slurm-Controllers zugegriffen werden. Um den externen Zugriff zu ermöglichen, implementieren Sie AWS Dienste wie Application Load Balancer mit VPC Link, API Gateway oder richten Sie VPC-Peering- oder VPN-Verbindungen für sichere Konnektivität ein.

Warum verwendet die API HTTP statt HTTPS?

Die Slurm-REST-API soll ein interner Endpunkt innerhalb des privaten Netzwerks Ihres Clusters sein. Für Produktionsbereitstellungen, die eine Verschlüsselung erfordern, können Sie die SSL/TLS Terminierung auf einer höheren Ebene in Ihrer Architektur implementieren, z. B. über ein API-Gateway, einen Load Balancer oder einen Reverse-Proxy.

Wie kontrolliere ich den Zugriff auf die REST-API?

Konfigurieren Sie die Sicherheitsgruppenregeln Ihres Clusters, um den Zugriff auf Port 6820 auf dem Slurm-Controller einzuschränken. Legen Sie Regeln für eingehenden Datenverkehr fest, um Verbindungen nur von vertrauenswürdigen IP-Bereichen oder bestimmten Quellen innerhalb Ihrer VPC zuzulassen und unbefugten Zugriff auf den API-Endpunkt zu blockieren.

Wie rotiere ich den JWT-Signaturschlüssel?

Versetzen Sie Ihren Cluster in den Wartungsmodus ohne aktive Instances und initiieren Sie dann die Schlüsselrotation über AWS Secrets Manager. Aktivieren Sie nach Abschluss der Rotation die Warteschlangen erneut. Alle vorhandenen JWT-Token werden ungültig und müssen mit dem neuen Signaturschlüssel von Secrets Manager neu generiert werden.

Muss die Slurm-Buchhaltung aktiviert sein, um die REST-API verwenden zu können?

Nein, die Slurm-Buchhaltung ist für grundlegende REST-API-Operationen wie die Einreichung und Überwachung von Jobs nicht erforderlich. Für den gesamten `/slurmdb` Endpunkt muss die Buchhaltung jedoch aktiv sein.

Welche Tools von Drittanbietern funktionieren mit der AWS PCS-REST-API?

Viele bestehende Slurm-REST-API-Clients sollten mit AWS PCS funktionieren, einschließlich Slurm Exporter für Prometheus und benutzerdefinierten Anwendungen SlurmWeb, die dem

Standard-Slurm-REST-API-Format folgen. Tools, die auf die Authentifizierung angewiesen sind, müssen jedoch geändert werden, damit sie mit `scontrol token` den PCS JWT-Anforderungen funktionieren. AWS

Fallen zusätzliche Kosten für die Nutzung der REST-API an?

Nein, es fallen keine zusätzlichen Gebühren für die Aktivierung oder Nutzung der Slurm-REST-API-Funktion an. Sie zahlen wie gewohnt nur für die zugrunde liegenden Cluster-Ressourcen.

Wie kann ich Fehler bei der REST-API beheben?

- Probleme mit der Netzwerkkonnektivität

  Wenn Sie den API-Endpunkt nicht erreichen können, werden bei HTTP-Anfragen an den Cluster-Controller Verbindungstimeouts oder Fehler „Verbindung verweigert“ angezeigt.

  Was zu tun ist: Stellen Sie sicher, dass sich Ihr Client in derselben VPC befindet oder über ein ordnungsgemäßes Netzwerk-Routing verfügt, und stellen Sie sicher, dass Ihre Sicherheitsgruppe HTTP-Verkehr auf Port 6820 von Ihrer Quell-IP oder Ihrem Quellsubnetz zulässt.

- Probleme mit der Slurm-REST-Authentifizierung

  Wenn Ihr JWT-Token ungültig, abgelaufen oder falsch signiert ist, geben API-Anfragen im Fehlerfeld der Antwort „Protokollauthentifizierungsfehler“ zurück.

  Beispiel für eine Fehlermeldung:

  ```
  {
  "errors": [
      {
      "description": "Batch job submission failed",
      "error_number": 1007,
      "error": "Protocol authentication error",
      "source": "slurm_submit_batch_job()"
      }
    ]
  }
  ```

  Was zu tun ist: Vergewissern Sie sich, dass Ihr JWT-Token richtig formatiert, nicht abgelaufen und mit dem richtigen Schlüssel von Secrets Manager signiert ist. Stellen Sie sicher, dass das Token korrekt formatiert ist und die erforderlichen Ansprüche enthält und dass Sie das richtige Authentifizierungs-Header-Format verwenden.

- Job kann nach dem Absenden nicht ausgeführt werden

  Wenn Ihr JWT-Token gültig ist, aber eine falsche interne Struktur oder einen falschen Inhalt enthält, sind Jobs möglicherweise in den Status pausiert (PD) mit einem Ursachencode übergegangen. `JobAdminHead` Verwenden Sie`scontrol show job <job-id>`, um den Job zu überprüfen — Sie werden sehen`JobState=PENDING, Reason=JobHeldAdmin`, und. `SystemComment=slurm_cred_create failure, holding job`

  Was zu tun ist: Die Hauptursache können falsche Werte in JWT sein. Stellen Sie sicher, dass das Token ordnungsgemäß strukturiert ist und die erforderlichen Ansprüche gemäß der PCS-Dokumentation enthält.

- Probleme mit Zugriffsrechten für Arbeitsverzeichnisse

  Wenn die in Ihrem JWT angegebene Benutzeridentität keine Schreibberechtigungen für das Arbeitsverzeichnis des Jobs hat, schlägt der Job mit Berechtigungsfehlern fehl, ähnlich wie bei der Verwendung `sbatch --chdir` mit einem Verzeichnis, auf das nicht zugegriffen werden kann.

  Was zu tun ist: Stellen Sie sicher, dass der in Ihrem JWT-Token angegebene Benutzer über die entsprechenden Berechtigungen für das Arbeitsverzeichnis des Jobs verfügt.

- Haben Sie immer noch Probleme?

  1. Lesen Sie die Dokumentation von SchedMD zur REST-API-Spezifikation.
  2. Ausführlichere Informationen zu Fehlern finden Sie in den Slurm-Controller-Protokollen (weitere Informationen finden Sie unterScheduler loggt sich in AWS PCS ein).

Compute-Knoten mit eingeschaltetem Slurm neu starten AWS STK.

AWS PCS unterstützt den nativen `scontrol reboot` Befehl von Slurm. Verwenden Sie diesen Befehl, um Rechenknoten neu zu starten, ohne die EC2-Instanz zu ersetzen. Andere Neustartmethoden (Amazon EC2 EC2-Konsole AWS CLI, automatische Patches oder Systemwartung) veranlassen AWS PCS, die EC2-Instance als fehlerhaft einzustufen und zu ersetzen.

Vorteile des Slurm-Neustarts

Der Slurm-Neustart bietet mehrere Vorteile für die Cluster-Wartung:

- Kapazität erhalten — Vermeiden Sie den Verlust von EC2-Instances mit beschränkter Kapazität an andere Kunden.
- Kosten senken — Vermeiden Sie unnötige Austauschzyklen für Instances und die fortgesetzte Abrechnung ungenutzter Knoten.
- Schnellere Wiederherstellung — Keine Verzögerungen bei der Bereitstellung im Vergleich zum Austausch von Instanzen.
- Betriebliche Flexibilität — Beseitigen Sie Speicherlecks, entfernen Sie temporäre Dateien und stellen Sie Knoten aus heruntergekommenen Zuständen wieder her.

Wann sollte Slurm Reboot verwendet werden

Verwenden Sie Slurm Reboot für allgemeine betriebliche Wartungsszenarien:

- Fehlerbehebung — Beheben Sie Leistungsprobleme oder nicht reagierende Prozesse, insbesondere bei GPU-Knoten.
- Säuberung von Ressourcen — Beseitigen Sie Speicherlecks, temporäre Dateien oder festgefahrene Prozesse`/tmp`, die die Arbeitsleistung beeinträchtigen.
- Wiederherstellung — Stellen Sie Knoten wieder her, wenn sie nicht mehr funktionieren oder heruntergefahren sind, bevor ein vollständiger Knotenaustausch erforderlich ist.

Einschränkungen

- Nur Slurm-Admin-Benutzer (Root-Benutzer) können Reboot-Befehle ausführen.
- Die Unterstützung für Neustarts ist auf `scontrol reboot` nur beschränkt.
- RebootProgram Konfiguration wird nicht unterstützt.
- Keine Konsolenschnittstelle — nur Befehlszeile.

Themen

Starten Sie einen Rechenknoten mit Slurm in neu AWS STK.

Verwenden Sie den nativen Reboot-Befehl von Slurm, um Leistungsprobleme zu lösen, Ressourcenprobleme zu beheben oder die Wiederherstellung aus heruntergefahrenen Zuständen ohne Verlust der EC2-Instance-Kapazität wiederherzustellen.

Voraussetzungen

- Slurm-Admin-Rechte (Root-Benutzerzugriff)
- Zugriff auf einen Login-Knoten im AWS PCS-Cluster

Verfahren

1. Stellen Sie über die EC2-Konsole eine Connect zu einem Anmeldeknoten her.
   a. Wählen Sie in der EC2-Konsole Instances aus.
   b. Wählen Sie Ihre Login-Node-Instance aus.
   c. Wählen Sie Connect aus.
2. Identifizieren Sie den Namen des Ziel-Compute-Knotens mit `sinfo` oder`scontrol show node`.

```
sinfo
# or
scontrol show node
```

3. Führen Sie den Befehl reboot mit einer der folgenden Optionen aus:

> **Warning**
>
> Nicht `nextstate=DOWN` zusammen mit dem `scontrol reboot` Befehl verwenden. Dieser Parameter kennzeichnet den Knoten als fehlerhaft und löst den Instanzersatz aus.

   - Grundlegender Neustart (wartet darauf, dass der Knoten inaktiv wird):

```
scontrol reboot nodename
```

   - Sofortiger Neustart (leert den Knoten und startet neu, wenn die Jobs abgeschlossen sind):

```
scontrol reboot ASAP nodename
```

- Mit folgendem Grund neu starten:

```
scontrol reboot ASAP reason="troubleshooting" nodename
```

- Neustart mit Wiederaufnahmestatus:

```
scontrol reboot ASAP nextstate=RESUME nodename
```

4. Überwachen Sie den Fortschritt des Neustarts mit`scontrol show node`.

```
scontrol show node nodename
```

5. Stellen Sie sicher, dass der Knoten nach Abschluss des Neustarts wieder betriebsbereit ist.

Brechen Sie einen ausstehenden Neustart ab in AWS STK.

Brechen Sie einen ausstehenden Neustart ab, um unnötige Ausfallzeiten zu vermeiden, wenn das Problem behoben wurde oder wenn ein Neustart nicht mehr erforderlich ist.

Voraussetzungen

- Slurm-Admin-Rechte
- Der Node muss einen ausstehenden Neustart haben (es wird der Status „Neustart ausgelöst“ angezeigt)
- Zugriff auf den Anmeldeknoten für die Befehlsausführung

Verfahren

1. Connect zum Login-Knoten her.
2. Stellen Sie sicher, dass für den Knoten ein Neustart ansteht, indem Sie`scontrol show node`.

```
scontrol show node nodename
```

   Suchen Sie im Knotenstatus nach „Neustart ausgelöst“.
3. Führen Sie den Befehl Cancel aus.

```
scontrol cancel_reboot nodename
```

4. Stellen Sie sicher, dass der Neustart abgebrochen wurde und der Knotenstatus wieder normal ist.

```
scontrol show node nodename
```

Häufig gestellte Fragen zum Slurm-Neustart in AWS STK.

Hier finden Sie Antworten auf häufig gestellte Fragen zur Verwendung von Slurm Reboot in AWS PCS.

Was ist die Unterstützung für den Neustart von Slurm?

Support für den nativen `scontrol reboot` Slurm-Befehl. Verwenden Sie diesen Befehl, um Rechenknoten ohne automatischen Instanzaustausch neu zu starten, wodurch die EC2-Instanzkapazität erhalten bleibt und die Betriebskosten gesenkt werden.

Wer kann Slurm-Reboot-Befehle verwenden?

Nur Slurm-Admin-Benutzer (Root-Benutzer) können Reboot-Befehle ausführen. Reguläre Benutzer, die versuchen, diese zu verwenden, `scontrol reboot` erhalten von Slurm die Fehlermeldung „Zugriff verweigert“, ohne dass dies Auswirkungen auf den Knoten hat.

Was passiert mit laufenden Jobs während eines Neustarts?

Standardmäßig werden Jobs vor dem Neustart normal abgeschlossen. Mit der ASAP-Option wird der Knoten entleert, um neue Jobs zu verhindern, und der Neustart erfolgt, nachdem die aktuellen Jobs abgeschlossen sind. Jobs können storniert oder für sofortige Neustarts in die Warteschlange gestellt werden.

Wie unterscheidet sich das vom Neustart der EC2-Konsole?

Beim Slurm-Neustart bleibt die EC2-Instance erhalten und ein Austausch wird vermieden, während bei Neustarts der EC2-Konsole PCS die Instance aufgrund fehlgeschlagener Integritätsprüfungen während des Neustarts ersetzt.

Kann ich benutzerdefinierte Neustart-Skripts konfigurieren?

Nein, die RebootProgram Konfiguration wird in der ersten Version nicht unterstützt. Die Funktion verwendet das standardmäßige Neustartverhalten von Slurm ohne Unterstützung für benutzerdefinierte Skripts.

Wie lange dauert ein Slurm-Neustart?

Die Neustartzeit hängt vom Instance-Typ, den Startprozessen des Kunden, der AMI-Konfiguration und davon ab, ob Jobs zuerst abgeschlossen werden müssen. Der Prozess umfasst das Warten auf den Abschluss von Jobs, den physischen Neustart, Integritätsprüfungen und die Registrierung des Slurmd-Daemons.

Kann ich den Verlauf der Neustarts einsehen?

Neustart-Ereignisse werden in Slurm-Logs (slurmctld und slurmd) aufgezeichnet, die überwacht werden können. CloudWatch Das Feld „Grund“ im Knotenstatus zeigt den Grund für den Neustart während des Vorgangs an.

Was passiert, wenn ein Knoten beim Neustart hängen bleibt?

Wenn ein Knoten den Neustartvorgang nicht innerhalb dieses Zeitraums ResumeTimeout abschließt, wird er als INAKTIV markiert. Überprüfen Sie die CloudWatch Protokolle auf Fehler, überprüfen Sie die Netzwerkkonnektivität und untersuchen Sie die Slurmd-Protokolle. Wenden Sie sich an den AWS Support, falls das Problem weiterhin besteht.

Kann ich mehrere Knoten gleichzeitig neu starten?

Ja, Sie können im Reboot-Befehl mehrere Knoten angeben:

```
scontrol reboot ASAP node1,node2,node3
```

Wie kann ich einen Knoten neu starten, ohne darauf zu warten, dass Jobs abgeschlossen sind?

Für sofortige Neustarts von Knoten bei Problemen wie problematischen Knoten, die sich auf Jobs mit mehreren Knoten auswirken, erheblichen Leistungseinbußen oder instabilem GPU-Verhalten haben Sie zwei Möglichkeiten:

- Abbrechen und neu starten — Brechen Sie zuerst die betroffenen Jobs mit `scancel <job_id>` ab und initiieren Sie dann einen sofortigen Neustart mit. `scontrol reboot ASAP <nodename>` Laufende Jobs werden beendet und müssen erneut eingereicht werden, nachdem der Knoten wiederhergestellt ist.

- Drain and Requeue (weniger wirksam) — Initiieren Sie zunächst einen Drain und starten Sie ihn neu. Stellen Sie dann die betroffenen Jobs mit `scontrol reboot ASAP <nodename>` in die Warteschlange. `scontrol requeue <job_id>` Dadurch werden Jobs wieder in den Status „Ausstehend“ versetzt, anstatt sie abzubrechen.

Was passiert, wenn ich nextState=down spezifiziere?

Wenn Sie dies angeben`nextstate=DOWN`, wird der Knoten nach dem Neustart als fehlerhaft markiert und löst den Instanzersatz aus. Um den Austausch von Instanzen zu vermeiden, geben Sie weder nextstate noch use an. `nextstate=RESUME`

Weitere Ressourcen

- Grundlegende Verfahren zum Neustart finden Sie unterStarten Sie einen Rechenknoten mit Slurm in neu AWS STK..
- Informationen zur Behebung von Neustartproblemen finden Sie unterBehebung von Problemen mit dem Slurm-Neustart in AWS STK..
- Die Dokumentation zum Slurm-Neustart finden Sie in der Slurm-Control-Dokumentation.

Behebung von Problemen mit dem Slurm-Neustart in AWS STK.

Wenn beim Neustart des Knotens Probleme auftreten, überprüfen Sie zunächst den Knotenstatus mithilfe von`scontrol show node` *`nodename`*. Untersuchen Sie dann die CloudWatch Protokolle sowohl auf Slurm (slurmctld und slurmd) als auch auf Systemprotokolle, um mögliche Fehler zu identifizieren.

Überprüfen Sie zur grundlegenden Fehlerbehebung die Netzwerkkonnektivität, überprüfen Sie die Sicherheitsgruppeneinstellungen und stellen Sie sicher, dass alle erforderlichen Dienste nach dem Neustart ausgeführt werden. Wenn die Probleme nach den grundlegenden Schritten zur Fehlerbehebung weiterhin bestehen, wenden Sie sich an den AWS Support. Wenn Sie sich an den Support wenden, geben Sie die entsprechenden Protokollauszüge, Informationen zum Knotenstatus und einen Zeitplan für den Neustartversuch an, um den Lösungsprozess zu beschleunigen.

Weitere Ressourcen

- Informationen zur Überwachung von AWS PCS-Instances mithilfe von CloudWatch Amazon finden Sie unter Überwachung von AWS PCS-Instances mithilfe von Amazon CloudWatch.

- Allgemeine Informationen zur Fehlerbehebung finden Sie unterBehebung von Problemen in AWS Dienst für parallele Datenverarbeitung.
- Die Slurm-Dokumentation finden Sie im Slurm-Leitfaden zur Fehlerbehebung.

Konfiguration benutzerdefinierter Slurm-Einstellungen in AWS STK.

Verwenden Sie benutzerdefinierte Slurm-Einstellungen, um zusätzliche Slurm-Parameter für Cluster-, Queue- und Compute Node Group-Ressourcen zu konfigurieren. Diese Version bietet Unterstützung für Slurm-Einstellungen für Queue-Ressourcen und bietet so eine detaillierte Kontrolle über partitionsspezifisches Verhalten.

Vorteile benutzerdefinierter Slurm-Einstellungen

Benutzerdefinierte Slurm-Einstellungen bieten eine ausgefeilte Kontrolle über Ihre AWS PCS-based HPC-Umgebung. Sie können eine detaillierte Abrechnung implementieren, Zugriffskontrollen durchsetzen und die Workload-Ausführung mithilfe von Quality-of-Service-Konfigurationen und Preemption-Richtlinien optimieren. Diese Funktionen stellen sicher, dass wichtige Aufgaben die erforderlichen Ressourcen erhalten und gleichzeitig eine effiziente Clusterauslastung gewährleistet wird. Ganz gleich, ob Sie GPU-accelerated Workloads verwalten, eine faire Planung implementieren oder die Lebenszyklen von Jobs kontrollieren — benutzerdefinierte Einstellungen helfen Ihnen dabei, Ihre HPC-Infrastruktur an den betrieblichen Anforderungen und Forschungszielen auszurichten.

Konfiguration benutzerdefinierter Einstellungen

Benutzerdefinierte Slurm-Einstellungen können während der Ressourcenerstellung über die AWS Konsole, CLI oder SDKs konfiguriert oder später durch Aktualisierungsvorgänge geändert werden.

AWS-Managementkonsole

Navigieren Sie auf der Erstellungs- oder Bearbeitungsseite für einen beliebigen Ressourcentyp (Cluster, Warteschlange oder Rechenknotengruppe) zu Zusätzliche Scheduler-Einstellungen.

Um eine neue Einstellung hinzuzufügen

1. Wählen Sie Neue Einstellung hinzufügen.
2. Wählen Sie einen Parameternamen aus der Dropdownliste aus (die kurze Parameterbeschreibungen enthält).
3. Geben Sie den entsprechenden Wert ein.

Um eine benutzerdefinierte Einstellung rückgängig zu machen

1. Wählen Sie neben dem entsprechenden parameter/value Paar die Option Entfernen aus.
2. Erstellen oder aktualisieren Sie die Ressource.

AWS CLI

Verwenden Sie das `SlurmCustomSettings` Feld für die programmatische Verwaltung benutzerdefinierter Einstellungen bei Erstellungs- oder Aktualisierungsvorgängen.

Example— Aktualisierung des Prolog-Parameters auf einem Cluster

```
aws pcs update-cluster --cluster-identifier my-cluster \
--slurm-configuration \
'SlurmCustomSettings=[{parameterName=Prolog,parameterValue="/path/to/prolog.sh"}]'
```

Example— Eine Warteschlange als Standard auf einem Cluster festlegen

```
aws pcs update-queue \
    --cluster-identifier my-cluster \
    --queue-identifier my-queue \
    --slurm-configuration
 'SlurmCustomSettings=[{parameterName=Default,parameterValue=YES}]'
```

Example— Einstellung benutzerdefinierter Funktionen für eine Compute-Knotengruppe

```
aws pcs update-compute-node-group \
    --cluster-identifier my-cluster \
    --compute-node-group-identifier my-cng-1 \
    --slurm-configuration \
    'SlurmCustomSettings=[{parameterName=Features,parameterValue="gpu,nvme"}]'
```

Validierung und Fehlerbehandlung

AWS PCS implementiert einen mehrstufigen Validierungsprozess für benutzerdefinierte Slurm-Einstellungen. Sowohl bei der Erstellung als auch bei der Aktualisierung führen wir synchrone Validierungen durch, die Folgendes beinhalten:

- Field-level Prüfungen: Wir überprüfen einzelne Einstellungen auf korrekte Datentypen, zulässige Werte und Formatanforderungen. Beispielsweise stellen wir sicher, dass Zeitwerte das richtige Slurm-Format haben und boolesche Werte akzeptierte boolesche Slurm-Repräsentationen verwenden.
- Context-aware Validierungen: Einige Einstellungen werden anhand des breiteren Konfigurationskontextes überprüft. Beispielsweise sind bestimmte Parameter nur gültig, wenn die Slurm-Buchhaltung aktiviert ist.
- Inter-setting Konsistenz: Wir stellen sicher, dass sich gegenseitig ausschließende Optionen nicht zusammen festgelegt wurden und dass die voneinander abhängigen Einstellungen korrekt konfiguriert sind.

Wenn die Überprüfung fehlschlägt, erhalten Sie eine `ValidationException` mit einem bestimmten Fehlercode (z. B. InvalidInput), einer eindeutigen Fehlermeldung, die das Problem beschreibt, und einer Liste der ungültigen Felder und ihrer jeweiligen Fehlerdetails.

Während dieser ersten Überprüfung werden zwar viele Probleme erkannt, einige komplexe Interaktionen zwischen den Einstellungen werden jedoch möglicherweise erst sichtbar, wenn die Konfiguration angewendet wird. In solchen Fällen schlägt der Vorgang mit einer informativen Fehlermeldung fehl, und alle teilweisen Änderungen werden rückgängig gemacht.

Einschränkungen

AWS PCS implementiert zum Schutz der Servicesicherheit und der Betriebsstabilität einen Ansatz mit Zulassungslisten. Einstellungen, die die Sicherheit von Dienstkonten gefährden oder die Funktionen verwalteter Dienste beeinträchtigen könnten, sind eingeschränkt. Wir evaluieren jedoch kontinuierlich die Kundenbedürfnisse und können auf der Grundlage von Kundenfeedback Unterstützung für weitere Einstellungen hinzufügen.

Themen

Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Cluster

Die folgenden benutzerdefinierten Slurm-Einstellungen werden auf Cluster-Ebene unterstützt:

- AccountingStorageEnforce
- AccountingStorageTRES
- AccountingStoreFlags
- AuthAltParameters
- CliFilterParameters

> **Note**
>
> Weitere Informationen zu CLI-Filtern finden Sie unterKonfiguration der Slurm-CLI-Filter-Plugins auf einem AWS PCS-Cluster. AWS PCS

- CliFilterPlugins

> **Note**
>
> Weitere Informationen zu CLI-Filtern finden Sie unterKonfiguration der Slurm-CLI-Filter-Plugins auf einem AWS PCS-Cluster. AWS PCS

- CommunicationParameters

> **Important**
>
> AWS PCS deaktiviert standardmäßig den HTTP-Endpunkt. Um ihn zu aktivieren, geben Sie an`enable_http`.

- DefMemPerCPU
- Epilog
- EnforcePartLimits
- FairShareDampeningFactor
- FirstJobId
- HealthCheckInterval
- HealthCheckNodeState

- HealthCheckProgram
- JobRequeue
- LaunchParameters
- Licenses
- MetricsType

> **Note**
>
> Weitere Informationen zu Metriken in AWS PCS finden Sie unterSlurm-Metriken in AWS STK..

- MinJobAge

> **Note**
>
> AWS PCS unterstützt einen Mindestwert von 5 Sekunden für`MinJobAge`.

- OverTimeLimit
- PreemptExemptTime
- PreemptMode
- PreemptParameters
- PreemptType
- PriorityCalcPeriod
- PriorityDecayHalfLife
- PriorityFavorSmall
- PriorityFlags
- PriorityMaxAge
- PriorityUsageResetPeriod
- PriorityWeightAge
- PriorityWeightAssoc
- PriorityWeightFairshare
- PriorityWeightJobSize
- PriorityWeightPartition

- PriorityWeightQOS
- PriorityWeightTRES
- Prolog
- PrologFlags
- RequeueExit
- RequeueExitHold
- SchedulerParameters
- SelectTypeParameters
- SrunPortRange
- TaskEpilog
- TaskPluginParam
- TaskProlog
- TrackWCKey
- UnkillableStepProgram
- UnkillableStepTimeout

Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Compute-Knotengruppen

Die folgenden benutzerdefinierten Slurm-Einstellungen werden auf Ebene der Compute-Knotengruppen unterstützt:

- CpuSpecList
- Features
- MemSpecLimit
- RealMemory
- Weight

Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Warteschlangen

Die folgenden benutzerdefinierten Slurm-Einstellungen werden auf Warteschlangenebene unterstützt:

- AllowAccounts

- AllowQos
- Default
- DefaultTime
- DenyAccounts
- DenyQos
- ExclusiveUser
- GraceTime
- MaxTime
- OverSubscribe
- OverTimeLimit
- PowerDownOnIdle
- PreemptMode
- PriorityJobFactor
- PriorityTier
- QOS
- TRESBillingWeights

Fehlerbehebung bei benutzerdefinierten Slurm-Einstellungen in AWS STK.

Wenn Sie beim Erstellen oder Aktualisieren von AWS PCS-Ressourcen mit benutzerdefinierten Slurm-Einstellungen auf Fehler stoßen, können Sie die Protokollierung verwenden, um die Probleme zu diagnostizieren und zu beheben.

Behebung inkompatibler benutzerdefinierter Slurm-Einstellungen

Problem: Sie erhalten eine Fehlermeldung, die der folgenden ähnelt, wenn Sie Cluster-, Compute-Knotengruppen- oder Warteschlangenoperationen ausführen:

```
{OPERATION} failed. The Slurm custom settings of the cluster might be incompatible.
 Check the settings and try again.
```

Dieser Fehler kann bei den folgenden Vorgängen auftreten:

- CreateCluster

- CreateComputeNodeGroup
- UpdateComputeNodeGroup
- CreateQueue
- UpdateQueue

Lösung: Aktivieren Sie die Protokollierung, um das spezifische Problem zu verstehen und die inkompatiblen Einstellungen zu beheben.

Zur Behebung inkompatibler benutzerdefinierter Slurm-Einstellungen

1. Erstellen Sie den Cluster, falls er noch nicht existiert, oder stellen Sie sicher, dass sich Ihr vorhandener Cluster in einem Zustand befindet, in dem die Protokollierung aktiviert werden kann.
2. Aktivieren Sie die Protokollierung für Ihren Cluster. Detaillierte Anweisungen finden Sie unter Protokollierung und Überwachung für AWS PCS.

   > **Note**
   >
   > Die Protokollierung kann aktiviert werden, sobald der Cluster erstellt wurde.

3. Überprüfen Sie die Protokolle, um das spezifische Problem mit der Slurm-Konfiguration zu identifizieren, das die Inkompatibilität verursacht hat.
4. Korrigieren Sie die inkompatiblen benutzerdefinierten Einstellungen anhand der Protokollinformationen und wiederholen Sie den Vorgang.

Informationen zu den unterstützten benutzerdefinierten Slurm-Einstellungen finden Sie unter:

- Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Cluster
- Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Compute-Knotengruppen
- Benutzerdefinierte Slurm-Einstellungen für AWS PCS-Warteschlangen

Konfiguration benutzerdefinierter Cgroup-Einstellungen in AWS STK.

Slurm verwendet das Linux-Cgroup-Subsystem, um Ressourcen für Jobs zu verwalten und einzuschränken, darunter Speicher, CPU-Kerne, Geräte und Auslagerungsspeicher. AWS Mit PCS

können Sie `cgroup.conf` Einstellungen auf Cluster-Ebene über die `CgroupCustomSettings` Eigenschaft „`SlurmConfiguration`Während der Clustererstellung oder -aktualisierung“ anpassen.

Konfiguration der Cgroup-Einstellungen

Benutzerdefinierte Cgroup-Einstellungen können während der Clustererstellung über die AWS Konsole, CLI oder SDKs konfiguriert oder später durch Aktualisierungsvorgänge geändert werden.

AWS-Managementkonsole

Navigieren Sie auf der Erstellungs- oder Bearbeitungsseite für eine Cluster-Ressource zu Zusätzliche Scheduler-Einstellungen.

Um eine neue Einstellung hinzuzufügen

1. Wählen Sie Neue Einstellung hinzufügen.
2. Wählen Sie einen Parameternamen aus der Dropdownliste aus (die kurze Parameterbeschreibungen enthält).
3. Geben Sie den entsprechenden Wert ein.

Um eine benutzerdefinierte Einstellung rückgängig zu machen

1. Wählen Sie neben dem entsprechenden parameter/value Paar die Option Entfernen aus.
2. Erstellen oder aktualisieren Sie die Ressource.

AWS CLI

Für die programmatische Verwaltung der Cgroup-Einstellungen verwenden Sie das `CgroupCustomSettings` Feld bei Clustererstellungs- oder Aktualisierungsvorgängen.

Example— Einstellung von ConstrainRamSpace auf einem Cluster

```
aws pcs update-cluster --cluster-identifier my-cluster \
--slurm-configuration \
'CgroupCustomSettings=[{parameterName=ConstrainRAMSpace,parameterValue="yes"}]'
```

Unterstützte Cgroup-Einstellungen für Cluster

Die folgenden benutzerdefinierten Cgroup-Einstellungen werden auf Clusterebene unterstützt:

- AllowedRAMSpace
- AllowedSwapSpace
- ConstrainCores
- ConstrainDevices
- ConstrainRAMSpace
- ConstrainSwapSpace
- IgnoreSystemd
- MaxRAMPercent
- MaxSwapPercent
- MinRAMSpace
- SignalChildrenProcesses

Konfiguration benutzerdefinierter SlurmDBD-Einstellungen in AWS STK.

Der Datenbank-Daemon von Slurm (slurmdbd) verwaltet Buchhaltungsdaten, Datenaufbewahrungsrichtlinien und Datenschutzkontrollen. AWS Mit PCS können Sie `slurmdbd.conf` Einstellungen auf Cluster-Ebene über die `SlurmdbdCustomSettings` Eigenschaft „`SlurmConfiguration`während der Cluster-Erstellung oder -Aktualisierung“ anpassen.

Konfiguration der Slurmdbd-Einstellungen

Benutzerdefinierte Slurmdbd-Einstellungen können während der Clustererstellung über die AWS Konsole, CLI oder SDKs konfiguriert oder später durch Aktualisierungsvorgänge geändert werden.

AWS-Managementkonsole

> Navigieren Sie auf der Erstellungs- oder Bearbeitungsseite für eine Cluster-Ressource zu Zusätzliche Scheduler-Einstellungen.

Um eine neue Einstellung hinzuzufügen

1. Wählen Sie Neue Einstellung hinzufügen.
2. Wählen Sie einen Parameternamen aus der Dropdownliste aus (die kurze Parameterbeschreibungen enthält).
3. Geben Sie den entsprechenden Wert ein.

Um eine benutzerdefinierte Einstellung rückgängig zu machen

1. Wählen Sie neben dem entsprechenden parameter/value Paar die Option Entfernen aus.
2. Erstellen oder aktualisieren Sie die Ressource.

AWS CLI

Verwenden Sie für die programmatische Verwaltung der Slurmdbd-Einstellungen das `SlurmdbdCustomSettings` Feld bei Clustererstellungs- oder -aktualisierungsvorgängen.

Example— TrackWCKey auf einem Cluster einrichten

```
aws pcs update-cluster --cluster-identifier my-cluster \
--slurm-configuration \
'SlurmdbdCustomSettings=[{parameterName=TrackWCKey,parameterValue="yes"}]'
```

Unterstützte slurmdbd-Einstellungen für Cluster

Die folgenden benutzerdefinierten Slurmdbd-Einstellungen werden auf Clusterebene unterstützt:

- AllowNoDefAcct
- AllResourcesAbsolute
- CommitDelay
- DefaultQOS
- MaxQueryTimeRange
- Parameters
- PrivateData
- PurgeEventAfter

- PurgeJobAfter
- PurgeResvAfter
- PurgeStepAfter
- PurgeSuspendAfter
- PurgeTXNAfter
- PurgeUsageAfter
- TrackWCKey

Erweitern Sie die Slurm-Funktionalität auf AWS PCS mit SPANK-Plugins

Verwenden Sie die SPANK-Plugins (Slurm Plug-in Architecture for Node and Job Kontrol), um das Verhalten von Slurm beim Starten und Ausführen von Jobs auf PCS-Clustern zu erweitern und zu ändern. AWS SPANK-Plugins bieten eine generische Schnittstelle zum Abfangen und Ändern von Jobstartphasen.

Installieren Sie SPANK-Plugins auf Ihrem Compute-Knoten-AMI und konfigurieren Sie sie, um das Verhalten Ihres Slurm-Clusters an Ihre Workload-Anforderungen anzupassen. Weitere Informationen zu SPANK finden Sie in der SPANK-Dokumentation auf der SchedMD-Website.

Inhalt

- Installieren Sie die SPANK-Plugins auf AWS STK.
- Konfiguriere SPANK-Plugins auf AWS STK.
- Häufig gestellte Fragen zu SPANK-Plugins auf AWS STK.

Installieren Sie die SPANK-Plugins auf AWS STK.

Folgen Sie der Plugin-Dokumentation, um SPANK-Plugins auf Ihrem AMI zu installieren.

Kompilieren Sie SPANK-Plugins für die spezifische Slurm-Version auf Ihrem Cluster. Das von AWS PCS bereitgestellte Slurm-Installationsprogramm speichert Slurm in. `/opt/aws/pcs/scheduler/slurm-`*`version`* Wenn Sie das Plugin kompilieren, geben Sie die Slurm-Version an.

Das folgende Beispiel zeigt, wie die Slurm-Version für einige Plugins angegeben wird:

```
export CFLAGS="-I/opt/aws/pcs/scheduler/slurm-version/include"
```

Wenn Sie mehrere Slurm-Versionen im AMI haben, kompilieren Sie das Plugin für jede Version. Speichern Sie die kompilierten Plugins in versionierten Ordnern.

Das folgende Beispiel zeigt, wie der Zielordner für einige Plugins angegeben wird:

```
export DESTDIR="your-preferred-versioned-path"
```

> **Important**
>
> Für Plugins sind möglicherweise unterschiedliche Variablen erforderlich. Weitere Informationen finden Sie in der offiziellen Dokumentation für das Plugin, das Sie installieren.

Konfiguriere SPANK-Plugins auf AWS STK.

Speichern Sie Konfigurationsdateien standardmäßig in`/etc/aws/pcs/scheduler/slurm-version/plugstack.conf.d/`.

Um Ihre SPANK-Konfiguration an einem anderen Ort zu speichern, fügen Sie Ihre Standorte zu einer Konfigurationsdatei im Standardverzeichnis hinzu.

Das folgende Beispiel zeigt, wie Sie Konfigurationsdateien aus anderen Verzeichnissen einbinden können:

```
# content of /etc/aws/pcs/scheduler/slurm-version/any-filename.conf
include path-to-your-configuration-folder/*.conf
include path-to-a-second-configuration-folder/*.conf
```

Speichern Sie jede Konfiguration in einer speziellen Datei oder in einer gemeinsamen Datei. Sie können mehrere Konfigurationsdateien verwenden.

Die folgenden Beispiele zeigen Beispielkonfigurationsdateien:

```
# content of path-to-your-or-default-config-folder/filename-1.conf
required path-to-plugin-1 arguments
optional path-to-plugin-2 arguments
```

```
# content of path-to-your-or-default-config-folder/filename-2.conf
required path-to-plugin-3 arguments
```

Weitere Informationen zur Konfiguration Ihrer Plugins finden Sie in der SPANK-Konfigurationsdokumentation auf der SchedMD-Website.

> **Important**
>
> Lege Ordnerberechtigungen fest, um unbefugte Änderungen an deiner Plugin-Konfiguration zu verhindern.

> **Note**
>
> AWS PCS verwaltet deine SPANK-Plugins nicht. Wenn Sie Fehler im Zusammenhang mit Plugins erhalten, überprüfen Sie die Fehlerprotokolle auf Ihren Rechenknoten.

> **Note**
>
> Slurm protokolliert fälschlicherweise einen Fehler ähnlich dem folgenden, wenn es deine SPANK-Konfiguration lädt:
>
> ```
> error: "Include" failed in file /etc/slurm/plugstack.conf line 3
> ```
>
> Sie können diesen Fehler ignorieren. Es hat keinen Einfluss darauf, wie SPANK-Plugins funktionieren.

Häufig gestellte Fragen zu SPANK-Plugins auf AWS STK.

In diesem Abschnitt werden häufig gestellte Fragen zur Installation und Konfiguration von SPANK-Plugins auf AWS PCS-Clustern behandelt.

Muss ich SPANK-Plugins sowohl auf Anmelde- als auch auf Rechenknoten installieren?

Einige SPANK-Plugins müssen nicht auf allen Knoten installiert werden. Aus Gründen der besseren Kompatibilität empfehlen wir jedoch, alle SPANK-Plugins auf jedem Knoten zu installieren.

Welche zusätzliche Konfiguration ist für den produktiven Einsatz von SPANK-Plugins erforderlich?

Neben der grundlegenden Installation und Konfiguration, die in den Beispielen gezeigt werden, erfordern Produktionsbereitstellungen in der Regel eine zusätzliche Einrichtung. Container-based Bei Plug-ins wie Pyxis müssen Sie möglicherweise Umgebungsvariablen für Enroot festlegen, PMI (Process Management Interface) aktivieren und Berechtigungen für die Container-Laufzeit konfigurieren. Die detaillierten Anforderungen für den Einsatz in der Produktion finden Sie in der Dokumentation des jeweiligen Plugins.

Wie behebe ich Probleme mit dem SPANK-Plugin?

AWS PCS verwaltet keine SPANK-Plugins. Untersuchen Sie die Fehlerprotokolle auf Ihren Rechenknoten, um Probleme zu beheben.

Verwenden Sie die Slurm CLI Filter Plugins, um die Einreichung von Jobs anzupassen in AWS STK.

AWS PCS unterstützt Slurm CLI Filter Plugins, um benutzerdefinierte Lua-Skripte auszuführen, die die Parameter für die Auftragsübermittlung auf Anmelde- und Rechenknoten validieren und ändern. Ausführliche Informationen zu CLI-Filter-Plugins finden Sie in der cli_filter-Plugin-API-Dokumentation auf der SchedMD-Website.

Voraussetzungen

CLI-Filter-Plugins erfordern Slurm Version 24.11 oder höher und ein Lua-Skript, das auf allen Anmelde- und Rechenknoten bereitgestellt wird.

> **Important**
>
> Für die Slurm-Versionen 24.11 und 25.05 erfordern die CLI Filter Plugins die Installation von Slurm mit dem AWS PCS Slurm-Installationsprogramm (Version 24.11.6-2+ oder 25.05.4-1+). Weitere Informationen zur Schritt 3 — Slurm installieren Installation von Slurm finden Sie unter.

Einschränkungen und Sicherheitsüberlegungen

- Durchsetzung der Sicherheit — CLI-Filter-Plugins können von jedem Benutzer leicht umgangen werden und dürfen nicht für sicherheitskritische Richtlinien verwendet werden. Benutzer können

CLI-Filter-Plugins deaktivieren, indem sie eine benutzerdefinierte Konfiguration angeben, die beim Senden von Jobs `CLIFilterPlugins` deaktiviert wurde.

- Nur Lua-Implementierung — Die Implementierung von Lua-Skripten wird unterstützt. Die C-Implementierung wird nicht unterstützt.

Themen

Konfiguration der Slurm-CLI-Filter-Plugins auf einem AWS PCS-Cluster

Konfigurieren Sie CLI-Filter-Plugins, wenn Sie einen neuen AWS PCS-Cluster erstellen. Sie können CLI-Filter-Plugins auf vorhandenen Clustern mithilfe der Update-API oder der Update-Konsole aktivieren oder deaktivieren, ohne den Cluster neu erstellen zu müssen.

Voraussetzungen

Führen Sie die folgenden Aufgaben aus, bevor Sie die CLI-Filter-Plug-ins konfigurieren:

- Stellen Sie sicher, dass Sie Slurm Version 24.11 oder höher verwenden. Informationen zu 24.11 und 25.05 finden Sie unter. the section called “Voraussetzungen”
- Schreiben und testen Sie ein Lua-Skript, das die CLI Filter Plugin API implementiert
- Benennen Sie Ihr Lua-Skript genau `cli_filter.lua`
- Wählen Sie eine Methode für die Bereitstellung Ihres Skripts auf allen Cluster-Instances (AMI, S3 oder Dateisystem)

Aktivieren Sie die CLI-Filter-Plugins auf einem neuen Cluster

AWS PCS console

1. Öffnen Sie die AWS PCS-Konsole unter https://console.aws.amazon.com/pcs/.
2. Klicken Sie im Navigationsbereich auf Cluster.

3. Wählen Sie Cluster erstellen.
4. Erweitern Sie unter Scheduler-Einstellungen die Option Zusätzliche Scheduler-Einstellungen.
5. Fügen Sie eine neue benutzerdefinierte Slurm-Einstellung hinzu, bei der der Parametername auf `CliFilterPlugins` und der Parameterwert auf eingestellt sind. `cli_filter/lua`
6. Schließen Sie die verbleibende Cluster-Konfiguration ab und wählen Sie Create cluster aus.

AWS PCS API

Geben Sie die `slurmCustomSettings` Konfiguration in Ihrem Aufruf der `CreateCluster` API-Aktion an. Stellen Sie „`parameterNameAn`“ `CliFilterPlugins` und „`parameterValueBiscli_filter/lua`“ ein. Weitere Informationen finden Sie CreateClusterin der AWS PCS-API-Referenz.

Das folgende Beispiel verwendet die AWS CLI , um die `CreateCluster` API-Aktion aufzurufen. Die benutzerdefinierte Einstellung `CliFilterPlugins=cli_filter/lua` aktiviert CLI-Filter-Plugins.

```
aws pcs create-cluster --cluster-name cluster-name \
--scheduler type=SLURM,version=25.11 \
--size SMALL \
--networking subnetIds=cluster-subnet-id,securityGroupIds=cluster-security-group-id
 \
--slurm-configuration \
'slurmCustomSettings=[{parameterName=CliFilterPlugins,parameterValue="cli_filter/
lua"}]'
```

CLI-Filter-Plugin-Skripts bereitstellen

So stellen Sie CLI-Filter-Plugin-Skripts in Ihrem Cluster bereit

1. Stellen Sie sicher, dass auf allen AMIs, die in Compute-Knotengruppen verwendet werden, Slurm über das AWS PCS Slurm-Installationsprogramm installiert ist.

> **Note**
>
> Wenn Sie das AWS PCS-Beispiel-AMI für alle Compute-Knotengruppen verwenden, überspringen Sie diesen Schritt. Slurm ist bereits installiert.

2. Stellen Sie Ihr `cli_filter.lua` Skript `/etc/aws/pcs/scheduler/slurm-<version>/cli_filter.lua` auf allen Instanzen im Cluster bereit.

   Zum Beispiel für Slurm Version 25.11:

   ```
   /etc/aws/pcs/scheduler/slurm-25.11/cli_filter.lua
   ```

3. Starten Sie alle Anmelde- und Rechenknoten mit Ihren vorbereiteten AMIs.
4. Testen Sie die Auftragsübermittlung, um sicherzustellen, dass das CLI-Filter-Plugin korrekt ausgeführt wird.

Aktivieren oder deaktivieren Sie CLI-Filter-Plugins auf vorhandenen Clustern

Sie können CLI-Filter-Plug-ins auf vorhandenen Clustern aktivieren oder deaktivieren, ohne Ihre Infrastruktur neu aufbauen zu müssen. Weitere Informationen finden Sie unter Aktualisierung eines Clusters in AWS PCS.

AWS PCS console

1. Öffnen Sie die AWS PCS-Konsole unter. https://console.aws.amazon.com/pcs/
2. Klicken Sie im Navigationsbereich auf Cluster.
3. Wählen Sie den zu aktualisierenden Cluster aus.
4. Wählen Sie Aktion bearbeiten aus.
5. Gehen Sie auf der Seite Cluster bearbeiten unter Zusätzliche Scheduler-Einstellungen wie folgt vor:

   - Um CLI-Filter-Plugins zu aktivieren: Fügen Sie eine neue benutzerdefinierte Slurm-Einstellung hinzu, bei der der Parametername auf `CliFilterPlugins` und der Parameterwert auf `cli_filter/lua` gesetzt sind.
   - Um CLI-Filter-Plugins zu deaktivieren: Entfernen Sie die vorhandene `CliFilterPlugins` Einstellung.

6. Wählen Sie Cluster aktualisieren, um die Änderungen zu übermitteln.
7. Überwachen Sie den Clusterstatus, der während des Vorgangs als „Aktualisierung“ und nach Abschluss der Aktualisierung als „Aktiv“ angezeigt wird.

AWS PCS API

Verwenden Sie die `UpdateCluster` API-Aktion, um CLI-Filter-Plugins zu aktivieren oder zu deaktivieren. Weitere Informationen finden Sie UpdateClusterin der AWS PCS-API-Referenz.

So aktivieren Sie CLI-Filter-Plugins auf einem vorhandenen Cluster:

```
aws pcs update-cluster --cluster-identifier my-cluster \
--slurm-configuration \
'slurmCustomSettings=[{parameterName=CliFilterPlugins,parameterValue="cli_filter/
lua"}]'
```

So deaktivieren Sie CLI-Filter-Plugins auf einem vorhandenen Cluster:

```
aws pcs update-cluster --cluster-identifier my-cluster \
--slurm-configuration \
'slurmCustomSettings=[]'
```

Erwartete Ergebnisse

Nachdem Sie die Konfiguration abgeschlossen haben:

- Ihr Cluster wurde mit aktiviertem CLI Filter Plugin erstellt
- Job-Eingaben lösen Ihre benutzerdefinierte Validierungslogik aus, bevor sie den Slurm-Controller erreichen
- Non-compliant Jobs werden mit Ihren benutzerdefinierten Fehlermeldungen abgelehnt
- Konforme Jobs werden normal über den Slurm-Scheduler abgewickelt

Fehlerbehebung

Das CLI-Filter-Plugin-Skript fehlt auf einem Knoten

Symptome: Die Auftragsübermittlung schlägt sofort mit einem Fehler beim Laden des Plugins fehl.

Wahrscheinliche Ursache: Das Skript wurde nicht für alle Instanzen bereitgestellt oder der Dateipfad oder der Name ist falsch.

Lösung: Stellen Sie sicher, dass das Skript auf allen Anmelde- und Rechenknoten mit dem exakten Dateinamen im richtigen Pfad vorhanden ist`cli_filter.lua`.

Ungültige Konfiguration des CLI-Filter-Plug-ins

Symptome: Die Clustererstellung schlägt mit einem Validierungsfehler fehl.

Wahrscheinliche Ursache: `CliFilterPlugins` Der Parameter wurde nicht auf das `cli_filter/lua` Format gesetzt.

Lösung: Verwenden Sie den exakten Parameterwert `cli_filter/lua` in`slurmCustomSettings`.

Verwenden Sie Amazon S3, um ein CLI-Filter-Plugin-Skript bereitzustellen in AWS STK.

Verwenden Sie S3, um Ihr CLI-Filter-Plugin-Skript bereitzustellen, wenn Sie die Logik für die Auftragsübermittlung auf einem Live-Cluster aktualisieren möchten, ohne AMIs neu erstellen zu müssen. Bei diesem Ansatz wird das Skript beim Start der Instanz mithilfe von Benutzerdaten von S3 heruntergeladen.

Voraussetzungen

Bevor Sie Ihr Skript mit S3 bereitstellen, führen Sie die folgenden Aufgaben aus:

- Stellen Sie sicher, dass Sie Slurm Version 24.11 oder höher verwenden. Informationen zu 24.11 und 25.05 finden Sie unter. the section called “Voraussetzungen”
- Erstellen Sie einen S3-Bucket mit Ihrem CLI Filter Plugin Lua-Skript
- Konfigurieren Sie das IAM-Instanzprofil mit Lesezugriff auf den S3-Bucket
- Richten Sie den S3 VPC-Gateway-Endpunkt für direkten Zugriff ohne Internet ein
- Bereiten Sie das Benutzerdatenskript für den Download von S3 vor

So stellen Sie das CLI Filter Plugin-Skript mit S3 bereit

1. Laden Sie Ihr `cli_filter.lua` Skript in Ihren S3-Bucket hoch.
2. Konfigurieren Sie Ihr IAM-Instanzprofil mit S3-Leseberechtigungen für den Bucket.
3. Fügen Sie Shell-Code zu den Benutzerdaten Ihrer Startvorlage hinzu, um das Skript herunterzuladen:

```
aws s3 cp s3://my-bucket/cli_filter.lua /etc/aws/pcs/scheduler/slurm-25.11/
cli_filter.lua
```

```
chmod 644 /etc/aws/pcs/scheduler/slurm-25.11/cli_filter.lua
```

4. Stellen Sie Rechenknotengruppen mit Ihren aktualisierten Startvorlagen bereit.
5. Testen Sie die Auftragsübermittlung, um die Skriptfunktionalität zu überprüfen.

Erwartete Ergebnisse

Nachdem Sie die S3-Bereitstellung abgeschlossen haben:

- Das CLI Filter Plugin-Skript wird beim Start automatisch auf alle Instanzen heruntergeladen
- Skriptaktualisierungen in S3 werden auf neu gestarteten Instances widergespiegelt
- Richtlinien für die Einreichung von Job werden im gesamten Cluster einheitlich durchgesetzt

Fehlerbehebung

S3-Zugriff verweigert

Symptome: Der Instanzstart schlägt fehl oder das Skript wurde nicht heruntergeladen.

Wahrscheinliche Ursache: Fehlende IAM-Berechtigungen oder fehlender S3-VPC-Endpunkt.

Lösung: Stellen Sie sicher, dass das IAM-Instanzprofil `s3:GetObject` berechtigt ist und der S3-VPC-Endpunkt konfiguriert ist.

Translate ein Slurm Job Submit-Plugin-Skript, um das CLI Filter Plugin in zu verwenden AWS STK.

Translate Sie Ihr vorhandenes Job Submit Plugin Lua-Skript in das CLI Filter Plugin, wenn Sie aus anderen Slurm-Umgebungen migrieren. Der Übersetzungsprozess beinhaltet die Aktualisierung von Funktionsnamen und Feldzugriffsmustern, damit sie mit der CLI Filter Plugin API funktionieren.

Voraussetzungen

Bevor Sie Ihr Skript übersetzen, führen Sie die folgenden Aufgaben aus:

- Überprüfe dein vorhandenes Lua-Skript für das Job Submit Plugin
- Verstehen Sie die Unterschiede zwischen den APIs Job Submit und CLI Filter Plugin
- Greifen Sie auf die Dokumentation zum Slurm CLI Filter Plugin zu

Um das Job Submit Plugin-Skript in das CLI Filter Plugin zu übersetzen

1. Überprüfen Sie Ihre vorhandenen Job Submit Plugin-Skriptfunktionen (`slurm_job_submit`,`slurm_job_modify`).
2. Identifizieren Sie äquivalente Funktionen des CLI-Filter-Plug-ins:
   - `slurm_job_submit`wird `slurm_cli_pre_submit`
   - `slurm_cli_setup_defaults`Zur Standardparametereinstellung hinzufügen
   - `slurm_cli_post_submit`Für Aktionen nach der Einreichung hinzufügen
3. Translate Sie die Jobvalidierungslogik von `job_desc` Feldern in den `options` Array-Zugriff:
   - `job_desc.account`wird `options["account"]`
   - `job_desc.partition`wird `options["partition"]`
   - `job_desc.features`wird `options["constraint"]`
4. Aktualisieren Sie die Protokollierung von Aufrufen von `slurm.log_user()` bis`slurm.log_error()`.
5. Testen Sie Ihr übersetztes Skript auf einem Entwicklungscluster.
6. Stellen Sie es auf Ihrem Produktionscluster gemäß dem standardmäßigen Bereitstellungsprozess für das CLI-Filter-Plug-In bereit.

Erwartete Ergebnisse

Nachdem Sie die Übersetzung abgeschlossen haben:

- Ihr übersetztes Skript bietet eine gleichwertige Bestätigung der Stelleneingabe
- Benutzern werden ähnliche Fehlermeldungen und Eingabeaufforderungen wie bei Ihrem ursprünglichen Job Submit Plugin angezeigt
- Die Richtlinien für die Einreichung von Job werden während der Migration zu AWS PCS beibehalten

Fehlerbehebung

Fehler bei der Skriptübersetzung

Symptome: Auftragsübermittlungen schlagen mit Lua-Ausführungsfehlern fehl.

Wahrscheinliche Ursache: Falscher Feldzugriff oder fehlerhafte Funktionsaufrufen im übersetzten Skript.

Lösung: Lesen Sie die API-Dokumentation für das CLI Filter Plugin und vergleichen Sie die Feldzuordnungen zwischen den Schnittstellen Job Submit und CLI Filter.

Häufig gestellte Fragen zu Slurm CLI Filter Plugins in AWS STK.

Lesen Sie diese häufig gestellten Fragen zu CLI-Filter-Plugins.

Was ist der Unterschied zwischen dem CLI Filter Plugin und dem Job Submit Plugin?

Das CLI Filter Plugin wird clientseitig auf Anmelde- und Rechenknoten ausgeführt, bevor die Jobübermittlung den Controller erreicht, während das Job Submit Plugin nach der Jobübermittlung serverseitig auf dem Controller ausgeführt wird. Das CLI-Filter-Plugin kann von Benutzern umgangen werden, hält aber keine Controller-Sperren fest. Job Submit ist zwar sicher, kann aber die Cluster-Leistung während der Ausführung beeinträchtigen.

Unterstützt AWS PCS das Slurm Job Submit Plugin?

Nein, das Job Submit Plugin wird in AWS PCS nicht unterstützt. Verwenden Sie stattdessen das CLI-Filter-Plugin für die Validierung und Änderung von Jobeinreichungen.

Kann ich das CLI Filter Plugin zur Durchsetzung von Sicherheitsvorkehrungen verwenden?

Nein, das CLI Filter Plugin kann von entschlossenen Benutzern umgangen werden und sollte nicht zur Durchsetzung von Sicherheitsvorkehrungen verwendet werden. Verwenden Sie es eher für Verbesserungen der Benutzererfahrung, für die Einstellung von Standardparametern und für Richtlinien als für sicherheitskritische Richtlinien.

Warum muss sich das Skript auf allen Rechenknoten befinden, nicht nur auf Anmeldeknoten?

Slurm-Befehle wie `srun` können innerhalb von Job-Skripten auf Rechenknoten ausgeführt werden, was auch die Ausführung des CLI Filter Plugins auslöst. Das Skript muss überall dort verfügbar sein, wo Slurm-Befehle ausgeführt werden.

Kann ich das CLI Filter Plugin-Skript auf einem Live-Cluster ändern?

Ja, wenn Sie den S3- oder Dateisystem-Bereitstellungsansatz verwenden. Neue Instances erhalten das aktualisierte Skript, für bestehende Instances muss das Skript jedoch manuell oder über die von Ihnen gewählte Bereitstellungsmethode aktualisiert werden.

Kann ich verschiedene CLI-Filter-Plugin-Skripte für verschiedene Compute-Knotengruppen verwenden?

Ja, aber das wird nicht empfohlen. Sie können Skripten mit unterschiedlicher Logik für verschiedene Compute-Knotengruppen bereitstellen, aber Sie sind dafür verantwortlich, Interdependenzen zu verwalten und Logiküberschneidungen zu vermeiden. Die meisten Kunden stellen einen Logiksatz für einen gesamten Cluster bereit.

Kann ich das CLI Filter Plugin mit C-Implementierung anstelle von Lua verwenden?

Die C-Implementierung wird nicht unterstützt. In AWS PCS wird nur die Implementierung von Lua-Skripten unterstützt. SchedMD empfiehlt Kunden, Lua über C zu verwenden, um die Implementierung von CLI-Filter-Plugins zu vereinfachen.

Kann ich das CLI Filter Plugin in einem vorhandenen Cluster ein- oder ausschalten?

Ja, Sie können das CLI-Filter-Plugin auf vorhandenen Clustern mithilfe der Update-API aktivieren oder deaktivieren, ohne den Cluster neu erstellen zu müssen.

Behebung von Problemen mit dem Slurm-CLI-Filter-Plugin in AWS STK.

Verwenden Sie diese Informationen zur Fehlerbehebung, um häufig auftretende Probleme mit dem CLI-Filter-Plugin zu beheben.

Die Jobübermittlung schlägt sofort mit einem Fehler beim Laden des Plugins fehl

Symptome: Benutzer erhalten beim Senden von Jobs Fehlermeldungen über das fehlende oder ausgefallene CLI-Filter-Plugin.

Mögliche Ursachen:

- Das CLI-Filter-Plugin-Skript fehlt auf einem oder mehreren Knoten
- Falscher Skriptdateiname (muss exakt sein`cli_filter.lua`)
- Das Skript wurde im falschen Verzeichnispfad bereitgestellt
- Das Skript hat falsche Dateiberechtigungen

Auflösung

- Stellen Sie sicher, dass das Skript `/etc/aws/pcs/scheduler/slurm-<version>/cli_filter.lua` auf allen Anmelde- und Rechenknoten vorhanden ist

- Überprüfen Sie, ob der Dateiname des Skripts exakt ist `cli_filter.lua`
- Stellen Sie sicher, dass das Skript über lesbare Berechtigungen verfügt (644 oder ähnlich)
- Testen Sie die Skriptbereitstellung auf einem einzelnen Anmeldeknoten, bevor Sie sie im vollständigen Cluster bereitstellen

Die Clustererstellung schlägt mit einem Validierungsfehler für das CLI Filter Plugin fehl

Symptome: Die Clustererstellung schlägt fehl und es wird ein Fehler wegen eines ungültigen `CliFilterPlugins` Parameters angezeigt.

Mögliche Ursachen:

- Falsches Parameterwertformat in `slurmCustomSettings`
- Tippfehler im Parameternamen oder -wert

Auflösung

- Verwenden Sie den exakten Parameternamen: `CliFilterPlugins`
- Verwenden Sie den exakten Parameterwert: `cli_filter/lua`
- Überprüfen Sie die JSON-Syntax im `slurmCustomSettings` Array

Das CLI-Filter-Plugin-Skript wird ausgeführt, aber die Jobvalidierung funktioniert nicht wie erwartet

Symptome: Jobs werden erfolgreich gesendet, aber die benutzerdefinierte Validierungslogik wird nicht ausgelöst oder führt zu unerwarteten Ergebnissen.

Mögliche Ursachen:

- Syntaxfehler im Lua-Skript
- Falsche Feldzugriffsmuster (Verwendung der Job Submit Plugin-Syntax anstelle des CLI Filter Plug-ins)
- Logikfehler in den Validierungsbedingungen

Auflösung

- Überprüfen Sie das Lua-Skript auf Syntaxfehler
- Stellen Sie sicher, dass der Feldzugriff das `options["field_name"]` Format anstelle von verwendet `job_desc.field_name`
- Fügen Sie Protokollierungsanweisungen zum Ausführungsablauf des Debug-Skripts hinzu
- Testen Sie zunächst die Skriptlogik mit einfachen Validierungsfällen

Die Bereitstellung von S3-Skripten schlägt

Symptome: Instanzen werden gestartet, aber das CLI-Filter-Plugin-Skript wird nicht von S3 heruntergeladen.

Mögliche Ursachen:

- Dem IAM-Instanzprofil fehlen S3-Leseberechtigungen
- S3-VPC-Endpunkt nicht konfiguriert
- Falscher S3-Bucket- oder Objektpfad in den Benutzerdaten

Auflösung

- Stellen Sie sicher, dass das IAM-Instanzprofil über die `s3:GetObject` Berechtigung für Ihren Bucket verfügt
- Konfigurieren Sie den S3 VPC-Gateway-Endpunkt für den direkten Zugriff
- Überprüfen Sie den S3-Bucket-Namen und den Objektpfad im Benutzerdatenskript
- Überprüfen Sie die Benutzerdatenprotokolle der Instanz auf Fehler beim Herunterladen von S3

Slurm-Metriken in AWS STK.

AWS PCS unterstützt die Metrikfunktion von Slurm, die Clusterdaten in Echtzeit über HTTP-Endpunkte bereitstellt, die mit Prometheus und anderen Überwachungssystemen kompatibel sind. Einzelheiten, einschließlich der Auswirkungen auf die Leistung und Sicherheitsaspekte, finden Sie im Metrics Guide in der Slurm-Dokumentation.

Voraussetzungen

Bevor Sie Slurm-Metriken aktivieren, stellen Sie sicher, dass Sie über Folgendes verfügen:

- Cluster-Version: Slurm-Version 25.11 oder höher.
- Sicherheitsgruppe: Regeln, die HTTP-Verkehr auf Port 6817 von Ihren gewünschten Quellen zulassen.

Aktivieren Sie den Metrik-Endpunkt

Legen Sie die folgenden benutzerdefinierten Slurm-Einstellungen auf Clusterebene fest:

- `MetricsType`— Muss ein unterstütztes Metrik-Plugin angeben, wie z. `metrics/openmetrics`

- `CommunicationParameters`— Muss enthaltenenable_http.

> **Important**
>
> Durch die Aktivierung `enable_http` wird ein nicht authentifizierter HTTP-Endpunkt verfügbar gemacht. Jeder mit Netzwerkzugriff auf Port 6817 kann Cluster-, Job- und Knotenmetriken lesen. Verwenden Sie Sicherheitsgruppenregeln, um den Zugriff nur auf vertrauenswürdige Quellen zu beschränken.

- `PrivateData`— Darf nicht gesetzt werden.

Weitere Informationen zu benutzerdefinierten Slurm-Einstellungen finden Sie unterKonfiguration benutzerdefinierter Slurm-Einstellungen in AWS STK..

Verwenden Sie den Metrik-Endpunkt

Fragen Sie den Metrik-Endpunkt von einem Host mit Netzwerkzugriff auf den Controller ab:

```
curl http://controller-ip:6817/metrics
```

Weitere Informationen zu verfügbaren Metriken und zur Scraping-Konfiguration finden Sie im Metrics Guide in der Slurm-Dokumentation.

Sicherheit bei AWS Dienst für parallele Datenverarbeitung

Cloud-Sicherheit AWS hat höchste Priorität. Als AWS Kunde profitieren Sie von Rechenzentren und Netzwerkarchitekturen, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame Verantwortung von Ihnen AWS und Ihnen. Das Modell der geteilten Verantwortung beschreibt dies als Sicherheit der Cloud und Sicherheit in der Cloud:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, auf der AWS Dienste in der ausgeführt AWS Cloud werden. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Third-party Prüfer testen und verifizieren regelmäßig die Wirksamkeit unserer Sicherheitsmaßnahmen im Rahmen der AWS . Weitere Informationen zu den Compliance-Programmen, die für AWS Parallel Computing Service gelten, finden Sie unter AWS Services im Umfang nach Compliance-Programm AWS .
- Sicherheit in der Cloud — Ihre Verantwortung richtet sich nach dem AWS Dienst, den Sie nutzen. Sie sind auch für andere Faktoren verantwortlich, etwa für die Vertraulichkeit Ihrer Daten, für die Anforderungen Ihres Unternehmens und für die geltenden Gesetze und Vorschriften.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Verwendung von AWS PCS anwenden können. In den folgenden Themen erfahren Sie, wie Sie AWS PCS so konfigurieren, dass Ihre Sicherheits- und Compliance-Ziele erreicht werden. Sie erfahren auch, wie Sie andere AWS Dienste nutzen können, die Sie bei der Überwachung und Sicherung Ihrer AWS PCS-Ressourcen unterstützen.

Topics

- Bewährte Sicherheitsmethoden für AWS Dienst für parallele Datenverarbeitung

Datenschutz in AWS Dienst für parallele Datenverarbeitung

Das AWS Modell der gilt für den Datenschutz in AWS Parallel Computing Service. Wie in diesem Modell beschrieben, AWS ist verantwortlich für den Schutz der globalen Infrastruktur, auf der alle Systeme laufen AWS Cloud. Sie sind dafür verantwortlich, die Kontrolle über Ihre in dieser Infrastruktur gehosteten Inhalte zu behalten. Sie sind auch für die Sicherheitskonfiguration und die Verwaltungsaufgaben für die von Ihnen verwendeten AWS-Services verantwortlich. Weitere Informationen zum Datenschutz finden Sie unter Häufig gestellte Fragen zum Datenschutz . Weitere Informationen zum Datenschutz in Europa finden Sie im Zentrum für die Datenschutz-Grundverordnung (DSGVO).

Aus Datenschutzgründen empfehlen wir, dass Sie AWS-Konto Anmeldeinformationen schützen und einzelne Benutzer mit AWS IAM Identity Center oder AWS Identity and Access Management (IAM) einrichten. So erhält jeder Benutzer nur die Berechtigungen, die zum Durchführen seiner Aufgaben erforderlich sind. Außerdem empfehlen wir, die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto die Multi-Faktor-Authentifizierung (MFA).
- Wird verwendet SSL/TLS , um mit AWS Ressourcen zu kommunizieren. Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein AWS CloudTrail. Informationen zur Verwendung von CloudTrail Pfaden zur Erfassung von AWS Aktivitäten finden Sie unter Arbeiten mit CloudTrail Pfaden im AWS CloudTrail Benutzerhandbuch.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen darin enthaltenen Standardsicherheitskontrollen AWS-Services.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu schützen.
- Wenn Sie für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine API FIPS 140-3-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter Federal Information Processing Standard (FIPS) 140-3.

Wir empfehlen dringend, in Freitextfeldern, z. B. im Feld Name, keine vertraulichen oder sensiblen Informationen wie die E-Mail-Adressen Ihrer Kunden einzugeben. Dies gilt auch, wenn Sie mit AWS PCS oder anderen Geräten arbeiten und die Konsole, die API oder SDKs AWS-Services verwenden.

AWS CLI AWS Alle Daten, die Sie in Tags oder Freitextfelder eingeben, die für Namen verwendet werden, können für Abrechnungs- oder Diagnoseprotokolle verwendet werden. Wenn Sie eine URL für einen externen Server bereitstellen, empfehlen wir dringend, keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL einzuschließen.

Verschlüsselung im Ruhezustand

Die Verschlüsselung ist standardmäßig für ruhende Daten aktiviert, wenn Sie einen AWS Parallel Computing Service (AWS PCS) -Cluster mit der AWS-Managementkonsole, AWS CLI, AWS PCS-API oder den AWS SDKs erstellen. AWS PCS verwendet einen AWS eigenen KMS-Schlüssel, um Daten im Ruhezustand zu verschlüsseln. Weitere Informationen finden Sie unter Kundenschlüssel und AWS Schlüssel im AWS KMS Entwicklerhandbuch. Sie können auch einen vom Kunden verwalteten Schlüssel verwenden. Weitere Informationen finden Sie unter Erforderliche KMS-Schlüsselrichtlinie für die Verwendung mit verschlüsselten EBS-Volumes in AWS STK..

Das Clustergeheimnis wird im von Secrets Manager verwalteten KMS-Schlüssel gespeichert AWS Secrets Manager und mit diesem verschlüsselt. Weitere Informationen finden Sie unter Arbeiten mit Clustergeheimnissen in AWS PCS.

In einem AWS PCS-Cluster werden die folgenden Daten gespeichert:

- Scheduler-Status — Er umfasst Daten zu laufenden Jobs und bereitgestellten Knoten im Cluster. Dies sind die Daten, die Slurm in den in Ihrem definierten Zustand beibehält. `StateSaveLocation` `slurm.conf` Weitere Informationen finden Sie in der Beschreibung von StateSaveLocationin der Slurm-Dokumentation. AWS PCS löscht Jobdaten, nachdem ein Job abgeschlossen ist.
- Scheduler Auth Secret — AWS PCS verwendet es, um die gesamte Scheduler-Kommunikation im Cluster zu authentifizieren.

Für Informationen zum Scheduler-Status verschlüsselt AWS PCS Daten und Metadaten automatisch, bevor sie in das Dateisystem geschrieben werden. Das verschlüsselte Dateisystem verwendet einen branchenüblichen AES-256 Verschlüsselungsalgorithmus für ruhende Daten.

Verschlüsselung während der Übertragung

Ihre Verbindungen zur AWS PCS-API verwenden die TLS-Verschlüsselung mit dem Signaturprozess von Signature Version 4, unabhängig davon, ob Sie die AWS Command Line Interface (AWS CLI) oder AWS SDKs verwenden. Weitere Informationen finden Sie im AWS Identity and Access

Management Benutzerhandbuch unter Signieren von AWS API-Anfragen. AWS verwaltet die Zugriffskontrolle über die API mit den IAM-Richtlinien für die Sicherheitsanmeldedaten, die Sie für die Verbindung verwenden.

AWS PCS verwendet TLS, um eine Verbindung zu anderen AWS Diensten herzustellen.

Innerhalb eines Slurm-Clusters ist der Scheduler mit dem `auth/slurm` Authentifizierungs-Plug-In konfiguriert, das die Authentifizierung für die gesamte Scheduler-Kommunikation ermöglicht. Slurm bietet keine Verschlüsselung auf Anwendungsebene für seine Kommunikation. Alle Daten, die über Cluster-Instances fließen, bleiben lokal in der EC2-VPC und unterliegen daher der VPC-Verschlüsselung, wenn diese Instances die Verschlüsselung bei der Übertragung unterstützen. Weitere Informationen finden Sie unter Verschlüsselung bei der Übertragung im Amazon Elastic Compute Cloud-Benutzerhandbuch. Die Kommunikation zwischen dem Controller (in einem Dienstkonto bereitgestellt) und den Clusterknoten in Ihrem Konto ist verschlüsselt.

Schlüsselverwaltung

AWS PCS verwendet einen AWS eigenen KMS-Schlüssel zum Verschlüsseln von Daten. Weitere Informationen finden Sie unter Kundenschlüssel und AWS Schlüssel im AWS KMS Entwicklerhandbuch. Sie können auch einen vom Kunden verwalteten Schlüssel verwenden. Weitere Informationen finden Sie unter Erforderliche KMS-Schlüsselrichtlinie für die Verwendung mit verschlüsselten EBS-Volumes in AWS STK..

Das Clustergeheimnis wird im von Secrets Manager verwalteten KMS-Schlüssel gespeichert AWS Secrets Manager und mit diesem verschlüsselt. Weitere Informationen finden Sie unter Arbeiten mit Clustergeheimnissen in AWS PCS.

Inter-network Datenschutz im Verkehr

AWS Die PCS-Rechenressourcen für einen Cluster befinden sich innerhalb einer VPC im Kundenkonto. Daher verbleibt der gesamte interne AWS PCS-Servicetraffic innerhalb eines Clusters im AWS Netzwerk und wird nicht über das Internet übertragen. Die Kommunikation zwischen dem Benutzer und den AWS PCS-Knoten kann über das Internet erfolgen. Wir empfehlen, SSH oder Systems Manager zu verwenden, um eine Verbindung zu den Knoten herzustellen. Weitere Informationen finden Sie unter Was ist AWS Systems Manager? im AWS Systems Manager Benutzerhandbuch.

Sie können auch die folgenden Angebote verwenden, um Ihr lokales Netzwerk zu AWS verbinden mit:

- AWS Site-to-Site VPN. Weitere Informationen finden Sie unter Was ist AWS Site-to-Site VPN? im AWS Site-to-Site VPN Benutzerhandbuch.
- Ein AWS Direct Connect. Weitere Informationen finden Sie unter Was ist AWS Direct Connect? im AWS Direct Connect Benutzerhandbuch.

Sie greifen auf die AWS PCS-API zu, um administrative Aufgaben für den Service auszuführen. Sie und Ihre Benutzer greifen auf die Slurm-Endpunktports zu, um direkt mit dem Scheduler zu interagieren.

API-Verkehr verschlüsseln

Um auf die AWS PCS-API zugreifen zu können, müssen Clients Transport Layer Security (TLS) 1.2 oder höher unterstützen. Wir benötigen TLS 1.2 und empfehlen TLS 1.3. Clients müssen auch Cipher Suites mit Perfect Forward Secrecy (PFS) wie Ephemeral (DHE) oder Elliptic Curve Ephemeral Diffie-Hellman (ECDHE) unterstützen. Diffie-Hellman Die meisten modernen Systemen wie Java 7 und höher unterstützen diese Modi. Außerdem müssen Anforderungen mit einer Zugriffsschlüssel-ID und einem geheimen Zugriffsschlüssel signiert sein, der einem IAM-Prinzipal zugeordnet ist. Sie können () auch verwenden, um temporäre Sicherheitsanmeldedaten zum Signieren von Anfragen zu generieren. AWS -Security-Token-Service AWS STS

Den Datenverkehr verschlüsseln

Die Verschlüsselung von Daten während der Übertragung wird von unterstützten EC2-Instances aus aktiviert, die auf den Scheduler-Endpunkt zugreifen, und zwischen ComputeNodeGroup Instances aus dem. AWS Cloud Weitere Informationen finden Sie unter Verschlüsselung während der Übertragung.

Erforderliche KMS-Schlüsselrichtlinie für die Verwendung mit verschlüsselten EBS-Volumes in AWS STK.

AWS PCS verwendet dienstbezogene Rollen, um Berechtigungen an andere zu delegieren. AWS-Services Die dienstgebundene AWS PCS-Rolle ist vordefiniert und umfasst Berechtigungen, die AWS PCS benötigt, um andere AWS-Services in Ihrem Namen anzurufen. Die vordefinierten Berechtigungen umfassen auch den Zugriff auf Ihre, Von AWS verwaltete Schlüssel aber nicht auf Ihre vom Kunden verwalteten Schlüssel.

In diesem Thema wird beschrieben, wie Sie die Schlüsselrichtlinie einrichten, die zum Starten von Instances erforderlich ist, wenn Sie einen vom Kunden verwalteten Schlüssel für die Amazon EBS-Verschlüsselung angeben.

> **Note**
>
> AWS PCS benötigt keine zusätzliche Autorisierung, um die Standardeinstellung Von AWS verwalteter Schlüssel zum Schutz der verschlüsselten Volumes in Ihrem Konto zu verwenden.

Inhalt

-Übersicht

Sie können Folgendes AWS KMS keys für die Amazon EBS-Verschlüsselung verwenden, wenn AWS PCS Instances startet:

- Von AWS verwalteter Schlüssel— Ein Verschlüsselungsschlüssel in Ihrem Konto, das Amazon EBS erstellt, besitzt und verwaltet. Dies ist der Standardverschlüsselungsschlüssel für ein neues Konto. Amazon EBS verwendet den Von AWS verwalteter Schlüssel für die Verschlüsselung, sofern Sie keinen vom Kunden verwalteten Schlüssel angeben.
- Kundenverwalteter Schlüssel – Ein benutzerdefinierter Verschlüsselungsschlüssel, den Sie erstellen, besitzen und verwalten. Weitere Informationen finden Sie unter Erstellen eines KMS-Schlüssels im AWS Key Management Service Entwicklerhandbuch.

> **Note**
>
> Der Schlüssel muss symmetrisch sein. Amazon EBS unterstützt keine asymmetrischen, vom Kunden verwalteten Schlüssel.

Sie konfigurieren vom Kunden verwaltete Schlüssel, wenn Sie verschlüsselte Snapshots oder eine Startvorlage erstellen, die verschlüsselte Volumes spezifiziert, oder wenn Sie die Verschlüsselung standardmäßig aktivieren.

Konfigurieren von Schlüsselrichtlinien

Ihre KMS-Schlüssel müssen über eine Schlüsselrichtlinie verfügen, die es AWS PCS ermöglicht, Instances mit Amazon EBS-Volumes zu starten, die mit einem vom Kunden verwalteten Schlüssel verschlüsselt sind.

Verwenden Sie die Beispiele auf dieser Seite, um eine Schlüsselrichtlinie zu konfigurieren, die AWS PCS Zugriff auf Ihren vom Kunden verwalteten Schlüssel gewährt. Sie können die Schlüsselrichtlinie des vom Kunden verwalteten Schlüssels bei der Erstellung des Schlüssels oder zu einem späteren Zeitpunkt ändern.

Die Schlüsselrichtlinie muss die folgenden Aussagen enthalten:

- Eine Anweisung, die es der im `Principal` Element angegebenen IAM-Identität ermöglicht, den vom Kunden verwalteten Schlüssel direkt zu verwenden. Sie umfasst Berechtigungen zur Ausführung der AWS KMS `Encrypt`,, `Decrypt` `ReEncrypt*` `GenerateDataKey*`, und `DescribeKey` -Operationen mit dem Schlüssel.
- Eine Anweisung, die es der im `Principal` Element angegebenen IAM-Identität ermöglicht, den `CreateGrant` Vorgang zum Generieren von Zuschüssen zu verwenden, die eine Teilmenge ihrer eigenen Berechtigungen an Personen delegieren AWS-Services , die in AWS KMS oder einen anderen Principal integriert sind. Auf diese Weise können sie den Schlüssel verwenden, um in Ihrem Namen verschlüsselte Ressourcen zu erstellen.

Ändern Sie keine vorhandenen Aussagen in der Richtlinie, wenn Sie die neuen Richtlinienerklärungen zu Ihrer wichtigsten Richtlinie hinzufügen.

Weitere Informationen finden Sie unter:

- create-key in der Befehlsreferenz AWS CLI
- put-key-policy in der Befehlsreferenz AWS CLI
- Finden Sie die Schlüssel-ID und den Schlüssel-ARN im AWS Key Management Service Entwicklerhandbuch
- Service-linked Rollen für AWS STK.
- Amazon EBS-Verschlüsselung im Amazon EBS-Benutzerhandbuch
- AWS Key Management Serviceim Entwicklerhandbuch AWS Key Management Service

Beispiel 1: Schlüsselrichtlinienabschnitte, welche Zugriff auf den kundenverwalteten Schlüssel erlauben

Fügen Sie der Schlüsselrichtlinie des vom Kunden verwalteten Schlüssels die folgenden Richtlinienerklärungen hinzu. Ersetzen Sie den Beispiel-ARN durch den ARN Ihrer `AWSServiceRoleForPCS` serviceverknüpften Rolle. Diese Beispielrichtlinie erteilt der serviceverknüpften Rolle (`AWSServiceRoleForPCS`) von AWS PCS die Berechtigung, den vom Kunden verwalteten Schlüssel zu verwenden.

```
{
   "Sid": "Allow service-linked role use of the customer managed key",
   "Effect": "Allow",
   "Principal": {
       "AWS": [
           "arn:aws:iam::account-id:role/aws-service-role/pcs.amazonaws.com/
AWSServiceRoleForPCS"
       ]
   },
   "Action": [
       "kms:Encrypt",
       "kms:Decrypt",
       "kms:ReEncrypt*",
       "kms:GenerateDataKey*",
       "kms:DescribeKey"
   ],
   "Resource": "*"
}
```

```
{
   "Sid": "Allow attachment of persistent resources",
```

```
    "Effect": "Allow",
    "Principal": {
        "AWS": [
            "arn:aws:iam::account-id:role/aws-service-role/pcs.amazonaws.com/
AWSServiceRoleForPCS"
        ]
    },
    "Action": [
        "kms:CreateGrant"
    ],
    "Resource": "*",
    "Condition": {
        "Bool": {
            "kms:GrantIsForAWSResource": true
        }
     }
}
```

Beispiel 2: Schlüsselrichtlinienabschnitte, welche Zugriff auf den kundenverwalteten Schlüssel über mehrere Konten erlauben

Wenn Sie einen vom Kunden verwalteten Schlüssel in einem anderen Konto als Ihrem AWS PCS-Cluster erstellen, müssen Sie einen Grant in Kombination mit der Schlüsselrichtlinie verwenden, um kontoübergreifenden Zugriff auf den Schlüssel zu ermöglichen.

Um Zugriff auf den Schlüssel zu gewähren

1. Fügen Sie der Schlüsselrichtlinie des vom Kunden verwalteten Schlüssels die folgenden Richtlinienerklärungen hinzu. Ersetzen Sie den Beispiel-ARN durch den ARN des anderen Kontos. *111122223333*Ersetzen Sie es durch die tatsächliche Konto-ID des Kontos AWS-Konto , in dem Sie den AWS PCS-Cluster erstellen möchten. Damit können Sie einem IAM-Benutzer oder einer IAM-Rolle im angegebenen Konto die Berechtigung erteilen, mit dem folgenden CLI-Befehl eine Berechtigung für den Schlüssel zu erstellen. Standardmäßig haben Benutzer keinen Zugriff auf den Schlüssel.

   ```
   {.
       "Sid": "Allow external account 111122223333 use of the customer managed key",
       "Effect": "Allow",
       "Principal": {
           "AWS": [
               "arn:aws:iam::111122223333:root"
   ```

```
        ]
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
}
```

```
{
    "Sid": "Allow attachment of persistent resources in external
 account 111122223333",
    "Effect": "Allow",
    "Principal": {
        "AWS": [
            "arn:aws:iam::111122223333:root"
        ]
    },
    "Action": [
        "kms:CreateGrant"
    ],
    "Resource": "*"
}
```

2. Erstellen Sie von dem Konto aus, in dem Sie den AWS PCS-Cluster erstellen möchten, einen Zuschuss, der die entsprechenden Berechtigungen an die mit dem AWS PCS-Dienst verknüpfte Rolle delegiert. Der Wert von `grantee-principal` ist der ARN der serviceverknüpften Rolle. Der Wert von `key-id` ist der ARN des Schlüssels.

   Das folgende Beispiel für den CLI-Befehl create-grant erteilt der im Konto genannten serviceverknüpften Rolle die *111122223333* Berechtigungen, den vom Kunden verwalteten Schlüssel `AWSServiceRoleForPCS` im Konto zu verwenden. *444455556666*

```
aws kms create-grant \
  --region us-west-2 \
  --key-id arn:aws:kms:us-
west-2:444455556666:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d \
  --grantee-principal arn:aws:iam::111122223333:role/aws-service-role/
pcs.amazonaws.com/AWSServiceRoleForPCS \
```

```
 --operations "Encrypt" "Decrypt" "ReEncryptFrom" "ReEncryptTo" "GenerateDataKey"
"GenerateDataKeyWithoutPlaintext" "DescribeKey" "CreateGrant"
```

> **Note**
>
> Der Benutzer, der die Anfrage stellt, muss über die erforderlichen Berechtigungen verfügen, um die Aktion verwenden zu können. `kms:CreateGrant`

Das folgende Beispiel für eine IAM-Richtlinie ermöglicht es einer IAM-Identität (Benutzer oder Rolle) in einem Konto, einen Zuschuss für das vom Kunden verwaltete Key-in-Konto *111122223333* zu erstellen. *444455556666*

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCreationOfGrantForTheKMSKeyinExternalAccount444455556666",
      "Effect": "Allow",
      "Action": "kms:CreateGrant",
      "Resource": "arn:aws:kms:us-
west-2:444455556666:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    }
  ]
}
```

Weitere Informationen über die Erstellung eines Zuschusses für einen KMS-Schlüssel in einem anderen AWS-Konto, finden Sie unter Berechtigungserteilungen in AWS KMS im AWS Key Management Service -Entwicklerhandbuch.

> **Important**
>
> Der Name der serviceverknüpften Rolle, der als Prinzipal des Empfängers angegeben wird, muss der Name einer vorhandenen Rolle sein. Um sicherzustellen, dass der Zuschuss AWS PCS die Verwendung des angegebenen KMS-Schlüssels ermöglicht,

sollten Sie die serviceverknüpfte Rolle nicht löschen und neu erstellen, nachdem Sie den Zuschuss erstellt haben.

Bearbeiten Sie die wichtigsten Richtlinien in der AWS KMS Konsole

Die Beispiele in den vorherigen Abschnitten zeigen nur, wie einer Schlüsselrichtlinie Anweisungen hinzugefügt werden, was nur eine Möglichkeit darstellt, eine Schlüsselrichtlinie zu ändern. Die einfachste Möglichkeit, eine Schlüsselrichtlinie zu ändern, besteht darin, die Standardansicht der AWS KMS Konsole für wichtige Richtlinien zu verwenden und eine IAM-Identität (Benutzer oder Rolle) zu einem der Hauptbenutzer für die entsprechende Schlüsselrichtlinie zu machen. Weitere Informationen finden Sie im AWS Key Management Service Entwicklerhandbuch unter Verwenden der AWS-Managementkonsole Standardansicht.

Warning

Die Standardansichtsrichtlinien der Konsole beinhalten Berechtigungen zur Ausführung von AWS KMS `Revoke` Vorgängen mit dem vom Kunden verwalteten Schlüssel. Wenn Sie eine Genehmigung widerrufen, mit der AWS-Konto Zugriff auf einen vom Kunden verwalteten Schlüssel in Ihrem Konto gewährt wurde, AWS-Konto verlieren die Benutzer in diesem Konto den Zugriff auf die verschlüsselten Daten und den Schlüssel.

Zugriff AWS Parallel Computing Service mithilfe eines Schnittstellenendpunkts (AWS PrivateLink)

Sie können AWS PrivateLink es verwenden, um eine private Verbindung zwischen Ihrer VPC und AWS Parallel Computing Service (AWS PCS) herzustellen. Sie können darauf zugreifen, AWS PCS als ob es in Ihrer VPC wäre, ohne ein Internet-Gateway, ein NAT-Gerät, eine VPN-Verbindung oder Direct Connect eine Verbindung zu verwenden. Instances in Ihrer VPC benötigen für den Zugriff AWS PCS keine öffentlichen IP-Adressen.

Sie stellen diese private Verbindung her, indem Sie einen Schnittstellen-Endpunkt erstellen, der von AWS PrivateLink unterstützt wird. Wir erstellen eine Endpunkt-Netzwerkschnittstelle in jedem Subnetz, das Sie für den Schnittstellen-Endpunkt aktivieren. Hierbei handelt es sich um vom Anforderer verwaltete Netzwerkschnittstellen, die als Eingangspunkt für den Datenverkehr dienen, der für AWS PCS bestimmt ist.

Weitere Informationen finden Sie AWS PrivateLink im AWS PrivateLink Leitfaden unter Zugriff AWS-Services durch.

Überlegungen zu AWS PCS

Bevor Sie einen Schnittstellenendpunkt für einrichten AWS PCS, lesen Sie den Artikel Zugriff auf einen AWS-Service mithilfe eines Schnittstellen-VPC-Endpunkts im AWS PrivateLink Handbuch.

AWS PCS unterstützt Aufrufe aller API-Aktionen über den Schnittstellenendpunkt.

Wenn Ihre VPC keinen direkten Internetzugang hat, müssen Sie einen VPC-Endpunkt konfigurieren, damit Ihre Compute-Knotengruppen-Instances die AWS PCS `RegisterComputeNodeGroupInstance`API-Aktion aufrufen können.

Erstellen Sie einen Schnittstellen-Endpunkt für AWS PCS

Sie können einen Schnittstellenendpunkt für die AWS PCS Verwendung entweder der Amazon VPC-Konsole oder der AWS Command Line Interface (AWS CLI) erstellen. Weitere Informationen finden Sie unter Erstellen eines Schnittstellenendpunkts im AWS PrivateLink -Leitfaden.

Erstellen Sie einen Schnittstellenendpunkt für die AWS PCS Verwendung des folgenden Servicenamens:

```
com.amazonaws.region.pcs
```

*`region`*Ersetzen Sie es durch die ID des AWS-Region , in dem der Endpunkt erstellt werden soll, z. `us-east-1` B.

Wenn Sie privates DNS für den Schnittstellenendpunkt aktivieren, können Sie API-Anfragen an die AWS PCS Verwendung des standardmäßigen regionalen DNS-Namens stellen. Beispiel, `pcs.us-east-1.amazonaws.com`.

Erstellen einer Endpunktrichtlinie für Ihren Schnittstellen-Endpunkt

Eine Endpunktrichtlinie ist eine IAM-Ressource, die Sie an einen Schnittstellen-Endpunkt anfügen können. Die standardmäßige Endpunktrichtlinie ermöglicht den vollen Zugriff AWS PCS über den Schnittstellenendpunkt. Um den Zugriff AWS PCS von Ihrer VPC aus zu kontrollieren, fügen Sie dem Schnittstellenendpunkt eine benutzerdefinierte Endpunktrichtlinie hinzu.

Eine Endpunktrichtlinie gibt die folgenden Informationen an:

- Die Prinzipale, die Aktionen ausführen können (AWS-Konten, IAM-Benutzer und IAM-Rollen).
- Aktionen, die ausgeführt werden können
- Die Ressourcen, auf denen die Aktionen ausgeführt werden können.

Weitere Informationen finden Sie unter Steuern des Zugriffs auf Services mit Endpunktrichtlinien im AWS PrivateLink -Leitfaden.

Beispiel: VPC-Endpunktrichtlinie für AWS PCS actions

Im Folgenden finden Sie ein Beispiel für eine benutzerdefinierte Endpunktrichtlinie. Wenn Sie diese Richtlinie an Ihren Schnittstellenendpunkt anhängen, gewährt sie allen Prinzipalen im Cluster Zugriff auf die aufgelisteten AWS PCS Aktionen mit den angegebenen. *cluster-id region*Ersetzen Sie es durch die ID AWS-Region des Clusters, z. B. `us-east-1` *account-id*Ersetzen Sie durch die AWS-Konto Nummer des Clusters.

```
{
    "Statement": [
            {
                "Action": [
                "pcs:CreateCluster",
                "pcs:ListClusters",
                "pcs:DeleteCluster",
                "pcs:GetCluster",
                ],
                "Effect": "Allow",
                "Principal": "*",
                "Resource": [
                    "arn:aws:pcs:region:account-id:cluster/cluster-id*"
                ]
            }
        ]
}
```

Identity and Access Management für AWS Dienst für parallele Datenverarbeitung

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service , den Zugriff auf Ressourcen sicher zu AWS kontrollieren. IAM-Administratoren kontrollieren, wer authentifiziert

(angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um PCS-Ressourcen zu verwenden AWS . IAM ist ein Programm AWS-Service , das Sie ohne zusätzliche Kosten nutzen können.

Themen

Zielgruppe

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von Ihrer Rolle ab:

- Servicebenutzer – Fordern Sie von Ihrem Administrator Berechtigungen an, wenn Sie nicht auf Features zugreifen können (siehe Fehlerbehebung AWS Identität und Zugriff auf den Parallel Computing Service).
- Serviceadministrator – Bestimmen Sie den Benutzerzugriff und stellen Sie Berechtigungsanfragen (siehe Wie AWS Parallel Computing Service funktioniert mit IAM).
- IAM-Administrator – Schreiben Sie Richtlinien zur Zugriffsverwaltung (siehe Identity-based Richtlinienbeispiele für AWS Dienst für parallele Datenverarbeitung).

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen sich als IAM-Benutzer authentifizieren oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich als föderierte Identität anmelden, indem Sie Anmeldeinformationen aus einer Identitätsquelle wie AWS IAM Identity Center (IAM Identity Center), Single Sign-On-Authentifizierung oder Anmeldeinformationen verwenden. Google/Facebook Weitere Informationen zum Anmelden finden Sie unter So melden Sie sich bei Ihrem AWS-Konto an im Benutzerhandbuch für AWS-Anmeldung .

AWS Bietet für den programmatischen Zugriff ein SDK und eine CLI zum kryptografischen Signieren von Anfragen. Weitere Informationen finden Sie unter AWS Signature Version 4 for API requests im IAM-Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, dem sogenannten AWS-Konto Root-Benutzer, der vollständigen Zugriff auf alle AWS-Services Ressourcen hat. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Eine Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter Tasks that require root user credentials im IAM-Benutzerhandbuch.

Verbundidentität

Es hat sich bewährt, dass menschliche Benutzer für den Zugriff AWS-Services mithilfe temporärer Anmeldeinformationen einen Verbund mit einem Identitätsanbieter verwenden müssen.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensverzeichnis, Ihrem Directory Service Web-Identitätsanbieter oder der AWS-Services mithilfe von Anmeldeinformationen aus einer Identitätsquelle zugreift. Verbundene Identitäten übernehmen Rollen, die temporäre Anmeldeinformationen bereitstellen.

Für die zentrale Zugriffsverwaltung empfehlen wir AWS IAM Identity Center. Weitere Informationen finden Sie unter Was ist IAM Identity Center? im AWS IAM Identity Center -Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein IAM-Benutzer ist eine Identität mit bestimmten Berechtigungen für eine einzelne Person oder Anwendung. Wir empfehlen die Verwendung temporärer Anmeldeinformationen anstelle von IAM-Benutzern mit langfristigen Anmeldeinformationen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter Erfordern, dass menschliche Benutzer den Verbund mit einem Identitätsanbieter verwenden müssen, um AWS mithilfe temporärer Anmeldeinformationen darauf zugreifen zu können.

Eine IAM-Gruppe spezifiziert eine Sammlung von IAM-Benutzern und erleichtert die Verwaltung von Berechtigungen für große Gruppen von Benutzern. Weitere Informationen finden Sie unter Anwendungsfälle für IAM-Benutzer im IAM-Benutzerhandbuch.

IAM-Rollen

Eine IAM-Rolle ist eine Identität mit spezifischen Berechtigungen, die temporäre Anmeldeinformationen bereitstellt. Sie können eine Rolle übernehmen, indem Sie von einer Benutzer- zu einer IAM-Rolle (Konsole) wechseln oder indem Sie eine AWS Oder-API-Operation AWS CLI aufrufen. Weitere Informationen finden Sie unter Methoden, um eine Rolle zu übernehmen im IAM-Benutzerhandbuch.

IAM-Rollen sind nützlich für den Verbundbenutzer-Zugriff, temporäre IAM-Benutzerberechtigungen, kontoübergreifenden Zugriff, serviceübergreifenden Zugriff und Anwendungen, die auf Amazon EC2 laufen. Weitere Informationen finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Verwalten des Zugriffs mit Richtlinien

Sie kontrollieren den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie definiert Berechtigungen, wenn sie mit einer Identität oder Ressource verknüpft sind. AWS bewertet diese Richtlinien, wenn ein Principal eine Anfrage stellt. Die meisten Richtlinien werden AWS als JSON-Dokumente gespeichert. Weitere Informationen zu JSON-Richtliniendokumenten finden Sie unter Übersicht über JSON-Richtlinien im IAM-Benutzerhandbuch.

Mit Hilfe von Richtlinien legen Administratoren fest, wer Zugriff auf was hat, indem sie definieren, welches Prinzipal welche Aktionen auf welchen Ressourcenund unter welchen Bedingungendurchführen darf.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator erstellt IAM-Richtlinien und fügt sie zu Rollen hinzu, die die Benutzer dann übernehmen können. IAM-Richtlinien definieren Berechtigungen unabhängig von der Methode, die zur Ausführung der Operation verwendet wird.

Identity-based Richtlinien

Identity-based Richtlinien sind Richtliniendokumente für JSON-Berechtigungen, die Sie an eine Identität (Benutzer, Gruppe oder Rolle) anhängen. Diese Richtlinien steuern, welche Aktionen Identitäten für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen

zum Erstellen identitätsbasierter Richtlinien finden Sie unter Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Identity-based Richtlinien können Inline-Richtlinien (direkt in eine einzelne Identität eingebettet) oder verwaltete Richtlinien (eigenständige Richtlinien, die mehreren Identitäten zugeordnet sind) sein. Informationen dazu, wie Sie zwischen verwalteten und Inline-Richtlinien wählen, finden Sie unter Choose between managed policies and inline policies im IAM-Benutzerhandbuch.

Resource-based Richtlinien

Resource-based Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anhängen. Beispiele hierfür sind Vertrauensrichtlinien für IAM-Rollen und Amazon S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben.

Resource-based Richtlinien sind Inline-Richtlinien, die sich in diesem Dienst befinden. Sie können AWS verwaltete Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

Weitere Richtlinientypen

AWS unterstützt zusätzliche Richtlinientypen, mit denen die maximalen Berechtigungen festgelegt werden können, die durch gängigere Richtlinientypen gewährt werden:

- Berechtigungsgrenzen – Eine Berechtigungsgrenze legt die maximalen Berechtigungen fest, die eine identitätsbasierte Richtlinie einer IAM-Entität erteilen kann. Weitere Informationen finden Sie unter Berechtigungsgrenzen für IAM-Entitäten im -IAM-Benutzerhandbuch.
- Service-Kontrollrichtlinien (SCPs) – SCPs legen die maximalen Berechtigungen für eine Organisation oder Organisationseinheit in AWS Organizations fest. Weitere Informationen finden Sie unter Service-Kontrollrichtlinien im AWS Organizations -Benutzerhandbuch.
- Ressourcen-Kontrollrichtlinien (RCPs) – RCPs definieren die maximale Anzahl an Berechtigungen, die Ressourcen in Ihren Konten zur Verfügung stehen. Weitere Informationen finden Sie unter Ressourcen-Kontrollrichtlinien im AWS Organizations -Benutzerhandbuch.
- Sitzungsrichtlinien – Sitzungsrichtlinien sind erweiterte Richtlinien, die als Parameter übergeben werden, wenn Sie eine temporäre Sitzung für eine Rolle oder einen Verbundbenutzer erstellen. Weitere Informationen finden Sie unter Sitzungsrichtlinien im IAM-Benutzerhandbuch.

Mehrere Richtlinientypen

Wenn mehrere Arten von Richtlinien für eine Anfrage gelten, sind die sich daraus ergebenden Berechtigungen schwieriger zu verstehen. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie unter Bewertungslogik für Richtlinien im IAM-Benutzerhandbuch.

Wie AWS Parallel Computing Service funktioniert mit IAM

Bevor Sie IAM zur Verwaltung des Zugriffs auf AWS PCS verwenden, sollten Sie sich darüber informieren, welche IAM-Funktionen für die Verwendung mit PCS verfügbar sind. AWS

IAM-Funktionen, die Sie mit verwenden können AWS Dienst für parallele Datenverarbeitung

IAM-Feature	AWS PCS-Unterstützung
Identity-based Richtlinien	Ja
Resource-based Richtlinien	Nein
Richtlinienaktionen	Ja
Richtlinienressourcen	Ja
Richtlinienbedingungsschlüssel (services pezifisch)	Ja
ACLs	Nein
ABAC (Tags in Richtlinien)	Ja
Temporäre Anmeldeinformationen	Ja
Prinzipalberechtigungen	Ja
Servicerollen	Nein
Service-linked Rollen	Ja

Einen allgemeinen Überblick darüber, wie AWS PCS und andere AWS Dienste mit den meisten IAM-Funktionen funktionieren, finden Sie im IAM-Benutzerhandbuch unter AWS Dienste, die mit IAM funktionieren.

Identity-based Richtlinien für AWS STK.

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identity-based Richtlinien sind Richtliniendokumente für JSON-Berechtigungen, die Sie an eine Identität anhängen können, z. B. an einen IAM-Benutzer, eine Benutzergruppe oder eine Rolle. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Mit identitätsbasierten IAM-Richtlinien können Sie angeben, welche Aktionen und Ressourcen zugelassen oder abgelehnt werden. Darüber hinaus können Sie die Bedingungen festlegen, unter denen Aktionen zugelassen oder abgelehnt werden. Informationen zu sämtlichen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie in der IAM-Referenz für JSON-Richtlinienelemente im IAM-Benutzerhandbuch.

Identity-based Richtlinienbeispiele für AWS STK.

Beispiele für identitätsbasierte Richtlinien von AWS PCS finden Sie unter. Identity-based Richtlinienbeispiele für AWS Dienst für parallele Datenverarbeitung

Resource-based Richtlinien innerhalb AWS STK.

Unterstützt ressourcenbasierte Richtlinien: Nein

Resource-based Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anhängen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Um kontoübergreifenden Zugriff zu ermöglichen, können Sie ein gesamtes Konto oder IAM-Entitäten in einem anderen Konto als Prinzipal in einer ressourcenbasierten Richtlinie angeben.

Weitere Informationen finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Richtlinienaktionen für AWS STK.

Unterstützt Richtlinienaktionen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien festlegen, wer auf was Zugriff hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Action` einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Nehmen Sie Aktionen in eine Richtlinie auf, um Berechtigungen zur Ausführung des zugehörigen Vorgangs zu erteilen.

Eine Liste der AWS PCS-Aktionen finden Sie unter Von AWS Parallel Computing Service definierte Aktionen in der Serviceautorisierungsreferenz.

Richtlinienaktionen in AWS PCS verwenden vor der Aktion das folgende Präfix:

```
pcs
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Kommata:

```
"Action": [
      "pcs:action1",
      "pcs:action2"
         ]
```

Politische Ressourcen für AWS STK.

Unterstützt Richtlinienressourcen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien festlegen, wer auf was Zugriff hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das JSON-Richtlinienelement `Resource` gibt die Objekte an, auf welche die Aktion angewendet wird. Als Best Practice geben Sie eine Ressource mit dem zugehörigen Amazon-Ressourcennamen (ARN) an. Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

```
"Resource": "*"
```

Eine Liste der AWS PCS-Ressourcentypen und ihrer ARNs finden Sie unter Von AWS Parallel Computing Service definierte Ressourcen in der Service Authorization Reference. Informationen darüber, mit welchen Aktionen Sie den ARN jeder Ressource angeben können, finden Sie unter Von AWS Parallel Computing Service definierte Aktionen.

Beispiele für identitätsbasierte AWS PCS-Richtlinien finden Sie unter. Identity-based Richtlinienbeispiele für AWS Dienst für parallele Datenverarbeitung

Bedingungsschlüssel für Richtlinien für AWS STK.

Unterstützt servicespezifische Richtlinienbedingungsschlüssel: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien festlegen, wer auf was Zugriff hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Condition` gibt an, wann Anweisungen auf der Grundlage definierter Kriterien ausgeführt werden. Sie können bedingte Ausdrücke erstellen, die Bedingungsoperatoren verwenden, z. B. ist gleich oder kleiner als, damit die Bedingung in der Richtlinie mit Werten in der Anforderung übereinstimmt. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontextschlüssel für AWS globale Bedingungen im IAM-Benutzerhandbuch.

Eine Liste der AWS PCS-Bedingungsschlüssel finden Sie unter Bedingungsschlüssel für AWS Parallel Computing Service in der Service Authorization Reference. Informationen zu den Aktionen und Ressourcen, mit denen Sie einen Bedingungsschlüssel verwenden können, finden Sie unter Von AWS Parallel Computing Service definierte Aktionen.

Beispiele für identitätsbasierte AWS PCS-Richtlinien finden Sie unter. Identity-based Richtlinienbeispiele für AWS Dienst für parallele Datenverarbeitung

ACLs in AWS STK.

Unterstützt ACLs: Nein

Zugriffssteuerungslisten (ACLs) steuern, welche Prinzipale (Kontomitglieder, Benutzer oder Rollen) auf eine Ressource zugreifen können. ACLs sind ähnlich wie ressourcenbasierte Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

ABAC mit AWS STK.

Unterstützt ABAC (Tags in Richtlinien): Ja

Attribute-based Access Control (ABAC) ist eine Autorisierungsstrategie, die Berechtigungen auf der Grundlage von Attributen definiert, die als Tags bezeichnet werden. Sie können Tags an IAM-Entitäten und AWS -Ressourcen anhängen und dann ABAC-Richtlinien entwerfen, die Operationen zulassen, wenn das Tag des Prinzipals mit dem Tag auf der Ressource übereinstimmt.

Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungselement einer Richtlinie Tag-Informationen an, indem Sie die Schlüssel `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, oder Bedingung `aws:TagKeys` verwenden.

Wenn ein Service alle drei Bedingungsschlüssel für jeden Ressourcentyp unterstützt, lautet der Wert für den Service Ja. Wenn ein Service alle drei Bedingungsschlüssel für nur einige Ressourcentypen unterstützt, lautet der Wert Teilweise.

Weitere Informationen zu ABAC finden Sie unter Definieren von Berechtigungen mit ABAC-Autorisierung im IAM-Benutzerhandbuch. Um ein Tutorial mit Schritten zur Einstellung von ABAC anzuzeigen, siehe Attributbasierte Zugriffskontrolle (ABAC) verwenden im IAM-Benutzerhandbuch.

Verwenden temporärer Anmeldeinformationen mit AWS STK.

Unterstützt temporäre Anmeldeinformationen: Ja

Temporäre Anmeldeinformationen ermöglichen kurzfristigen Zugriff auf AWS Ressourcen und werden automatisch erstellt, wenn Sie den Verbund verwenden oder die Rollen wechseln. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter Temporäre Anmeldeinformationen in IAM und AWS-Services , die mit IAM funktionieren im IAM-Benutzerhandbuch.

Cross-service Hauptberechtigungen für AWS STK.

Unterstützt Forward Access Sessions (FAS): Ja

Forward Access Sessions (FAS) verwenden die Berechtigungen des Principals, der einen aufruft AWS-Service, in Kombination mit der Anfrage, Anfragen AWS-Service an nachgelagerte Dienste zu stellen. Einzelheiten zu den Richtlinien für FAS-Anforderungen finden Sie unter Zugriffssitzungen weiterleiten.

Servicerollen für AWS STK.

Unterstützt Servicerollen: Nein

Eine Servicerolle ist eine IAM-Rolle, die ein Service annimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service im IAM-Benutzerhandbuch.

> **Warning**
>
> Durch das Ändern der Berechtigungen für eine Servicerolle kann die Funktionalität von AWS PCS beeinträchtigt werden. Bearbeiten Sie Servicerollen nur, wenn AWS PCS Sie dazu anleitet.

Service-linked Rollen für AWS STK.

Unterstützt serviceverknüpfte Rollen: Ja

Eine dienstbezogene Rolle ist eine Art von Servicerolle, die mit einer AWS-Service verknüpft ist. Der Dienst kann die Rolle übernehmen, eine Aktion in Ihrem Namen auszuführen. Service-linked Rollen erscheinen in Ihrem Dienst AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

Einzelheiten zum Erstellen oder Verwalten von Rollen, die mit dem AWS PCS-Dienst verknüpft sind, finden Sie unterService-linked Rollen für AWS STK..

Identity-based Richtlinienbeispiele für AWS Dienst für parallele Datenverarbeitung

Standardmäßig sind Benutzer und Rollen nicht berechtigt, AWS PCS-Ressourcen zu erstellen oder zu ändern. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter Erstellen von IAM-Richtlinien (Konsole) im IAM-Benutzerhandbuch.

Einzelheiten zu den von AWS PCS definierten Aktionen und Ressourcentypen, einschließlich des Formats der ARNs für die einzelnen Ressourcentypen, finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für AWS Parallel Computing Service in der Service Authorization Reference.

Themen

- Best Practices für Richtlinien
- Verwendung der AWS PCS-Konsole
- Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

Best Practices für Richtlinien

Identity-based Richtlinien legen fest, ob jemand AWS PCS-Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder diese löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Beachten Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Richtlinien und Empfehlungen:

- Erste Schritte mit AWS verwalteten Richtlinien und Umstellung auf Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um damit zu beginnen, Ihren Benutzern und Workloads Berechtigungen zu gewähren. Sie sind in Ihrem verfügbar. AWS-Konto Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom AWS Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind. Weitere Informationen finden Sie unter Von AWS verwaltete Richtlinien oder Von AWS verwaltete Richtlinien für Auftragsfunktionen im IAM-Benutzerhandbuch.
- Anwendung von Berechtigungen mit den geringsten Rechten – Wenn Sie mit IAM-Richtlinien Berechtigungen festlegen, gewähren Sie nur die Berechtigungen, die für die Durchführung einer Aufgabe erforderlich sind. Sie tun dies, indem Sie die Aktionen definieren, die für bestimmte Ressourcen unter bestimmten Bedingungen durchgeführt werden können, auch bekannt als die geringsten Berechtigungen. Weitere Informationen zur Verwendung von IAM zum Anwenden von Berechtigungen finden Sie unter Richtlinien und Berechtigungen in IAM im IAM-Benutzerhandbuch.
- Verwenden von Bedingungen in IAM-Richtlinien zur weiteren Einschränkung des Zugriffs – Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen zu beschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben, um festzulegen, dass alle Anforderungen mithilfe von SSL gesendet werden müssen. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese für einen

bestimmten Zweck verwendet werden AWS-Service, z. CloudFormation B. Weitere Informationen finden Sie unter IAM-JSON-Richtlinienelemente: Bedingung im IAM-Benutzerhandbuch.

- Verwenden von IAM Access Analyzer zur Validierung Ihrer IAM-Richtlinien, um sichere und funktionale Berechtigungen zu gewährleisten – IAM Access Analyzer validiert neue und vorhandene Richtlinien, damit die Richtlinien der IAM-Richtliniensprache (JSON) und den bewährten IAM-Methoden entsprechen. IAM Access Analyzer stellt mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen zur Verfügung, damit Sie sichere und funktionale Richtlinien erstellen können. Weitere Informationen finden Sie unter Richtlinienvalidierung mit IAM Access Analyzer im IAM-Benutzerhandbuch.
- Multi-Faktor-Authentifizierung (MFA) erforderlich — Wenn Sie ein Szenario haben, das IAM-Benutzer oder einen Root-Benutzer in Ihrem System erfordert AWS-Konto, aktivieren Sie MFA für zusätzliche Sicherheit. Um MFA beim Aufrufen von API-Vorgängen anzufordern, fügen Sie Ihren Richtlinien MFA-Bedingungen hinzu. Weitere Informationen finden Sie unter Sicherer API-Zugriff mit MFA im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter Best Practices für die Sicherheit in IAM im IAM-Benutzerhandbuch.

Verwendung der AWS PCS-Konsole

Um auf die AWS Parallel Computing Service-Konsole zugreifen zu können, benötigen Sie ein Mindestmaß an Berechtigungen. Diese Berechtigungen müssen es Ihnen ermöglichen, Details zu den AWS PCS-Ressourcen in Ihrem aufzulisten und anzuzeigen AWS-Konto. Wenn Sie eine identitätsbasierte Richtlinie erstellen, die strenger ist als die mindestens erforderlichen Berechtigungen, funktioniert die Konsole nicht wie vorgesehen für Entitäten (Benutzer oder Rollen) mit dieser Richtlinie.

Sie müssen Benutzern, die nur die API AWS CLI oder die AWS API aufrufen, keine Mindestberechtigungen für die Konsole gewähren. Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die der API-Operation entsprechen, die die Benutzer ausführen möchten.

Weitere Informationen zu den Mindestberechtigungen, die für die Verwendung der AWS PCS-Konsole erforderlich sind, finden Sie unterMindestberechtigungen für AWS STK..

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die IAM-Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer

Benutzeridentität angefügt sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der API AWS CLI oder AWS .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS verwaltete Richtlinien für AWS Dienst für parallele Datenverarbeitung

Eine AWS verwaltete Richtlinie ist eine eigenständige Richtlinie, die von erstellt und verwaltet wird AWS. AWS Verwaltete Richtlinien sind so konzipiert, dass sie Berechtigungen für viele gängige Anwendungsfälle bereitstellen, sodass Sie damit beginnen können, Benutzern, Gruppen und Rollen Berechtigungen zuzuweisen.

Beachten Sie, dass AWS verwaltete Richtlinien für Ihre speziellen Anwendungsfälle möglicherweise keine Berechtigungen mit den geringsten Rechten gewähren, da sie für alle AWS Kunden verfügbar sind. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind.

Sie können die in AWS verwalteten Richtlinien definierten Berechtigungen nicht ändern. Wenn die in einer AWS verwalteten Richtlinie definierten Berechtigungen AWS aktualisiert werden, wirkt sich das Update auf alle Prinzidentitäten (Benutzer, Gruppen und Rollen) aus, denen die Richtlinie zugeordnet ist. AWS aktualisiert eine AWS verwaltete Richtlinie höchstwahrscheinlich, wenn eine neue Richtlinie eingeführt AWS-Service wird oder neue API-Operationen für bestehende Dienste verfügbar werden.

Weitere Informationen finden Sie unter Von AWS verwaltete Richtlinien im IAM-Benutzerhandbuch.

AWS verwaltete Richtlinie: AWSPCSComputeNodePolicy

Sie können eine Verbindung AWSPCSComputeNodePolicy zu Ihren IAM-Entitäten herstellen. Sie können diese Richtlinie an eine IAM-Rolle für AWS PCS-Rechenknoten anhängen, die Sie angeben, damit Knoten, die diese Rolle verwenden, eine Verbindung zu einem AWS PCS-Cluster herstellen können.

AWS PCS ordnet diese Richtlinie einer Compute-Knotengruppenrolle zu, wenn Sie die Konsole verwenden, um eine Compute-Knotengruppe zu erstellen.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `pcs:RegisterComputeNodeGroupInstance`— Erlaubt einem AWS PCS-Rechenknoten (EC2-Instance), sich bei einem AWS PCS-Cluster zu registrieren.

Informationen zu den Berechtigungen für diese Richtlinie finden Sie unter AWSPCSComputeNodePolicy in der Referenz zu von AWS verwalteten Richtlinien.

AWS verwaltete Richtlinie: AWSPCSServiceRolePolicy

Sie können keine Verbindungen AWSPCSServiceRolePolicy zu Ihren IAM-Entitäten herstellen. Diese Richtlinie ist mit einer dienstbezogenen Rolle verknüpft, die es AWS PCS ermöglicht, Aktionen in Ihrem Namen durchzuführen. Weitere Informationen finden Sie unter Service-linked Rollen für AWS STK..

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `ec2`— Ermöglicht AWS PCS die Erstellung und Verwaltung von Amazon EC2 EC2-Ressourcen.
- `iam`— Ermöglicht AWS PCS, eine servicebezogene Rolle für die Amazon EC2-Flotte zu erstellen und die Rolle an Amazon EC2 weiterzugeben.
- `cloudwatch`— Ermöglicht AWS PCS die Veröffentlichung von Servicemetriken auf Amazon CloudWatch.
- `secretsmanager`— Ermöglicht AWS PCS die Verwaltung von Geheimnissen für AWS PCS-Clusterressourcen.

Informationen zu den Berechtigungen für diese Richtlinie finden Sie unter AWSPCSServiceRolePolicy in der Referenz zu von AWS verwalteten Richtlinien.

AWS PCS aktualisiert auf AWS Verwaltete Richtlinien

Hier finden Sie Informationen zu Aktualisierungen der AWS verwalteten Richtlinien für AWS PCS, seit dieser Dienst begonnen hat, diese Änderungen nachzuverfolgen. Abonnieren Sie den RSS-Feed auf der Seite AWS PCS Document History, um automatische Benachrichtigungen über Änderungen an dieser Seite zu erhalten.

Änderungen	Beschreibung	Datum
AWSPCSServiceRolePolicy – Aktualisierung auf eine bestehende Richtlinie	AWS PCS hat neue Berechtigungen zur Unterstützung von Kapazitätsblöcken für vorhersehbare Rechenkapazität hinzugefügt.	11. September 2025

Änderungen	Beschreibung	Datum
	Es wurde die `ec2:DescribeCapacityReservations` Berechtigung hinzugefügt, mit der AWS PCS Kapazitätsblockreservierungen für Rechenknotengruppen ermitteln und verwenden kann.	
AWSPCSComputeNodePolicy – Neue Richtlinie	AWS PCS hat eine neue Richtlinie hinzugefügt, um AWS PCS-Rechenknoten die Erlaubnis zu erteilen, sich mit AWS PCS-Clustern zu verbinden. AWS PCS ordnet diese Richtlinie einer IAM-Rolle zu, wenn Sie eine Rechenkno tengruppe in der AWS PCS-Konsole erstellen.	23. Juni 2025
Die JSON-Datei in diesem Dokument wurde aktualisiert	Der JSON-Code in diesem Dokument wurde korrigiert und enthält nun`"arn:aws:ec2:*:*:spot-instances-request/*"` .	5. September 2024
AWS PCS hat begonnen, Änderungen zu verfolgen	AWS PCS begann, Änderungen für seine AWS verwalteten Richtlinien nachzuverfolgen.	28. August 2024

Service-linked Rollen für AWS STK.

AWS Parallel Computing Service verwendet dienstgebundene AWS Identity and Access Management Rollen (IAM). Eine dienstgebundene Rolle ist ein einzigartiger Typ von IAM-Rolle,

die direkt mit PCS verknüpft ist. AWS Service-linked Rollen sind von AWS PCS vordefiniert und beinhalten alle Berechtigungen, die der Dienst benötigt, um andere AWS Dienste in Ihrem Namen aufzurufen.

Eine dienstbezogene Rolle erleichtert die Einrichtung von AWS PCS, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. AWS PCS definiert die Berechtigungen seiner dienstbezogenen Rollen, und sofern nicht anders definiert, kann nur AWS PCS seine Rollen übernehmen. Die definierten Berechtigungen umfassen die Vertrauensrichtlinie und die Berechtigungsrichtlinie, und diese Berechtigungsrichtlinie kann keiner anderen juristischen Stelle von IAM zugeordnet werden.

Sie können eine serviceverknüpfte Rolle erst löschen, nachdem die zugehörigen Ressourcen gelöscht wurden. Dadurch werden Ihre AWS PCS-Ressourcen geschützt, da Sie nicht versehentlich die Zugriffsberechtigung für die Ressourcen entziehen können.

Informationen zu anderen Diensten, die dienstbezogene Rollen unterstützen, finden Sie unter AWS Dienste, die mit IAM funktionieren. Suchen Sie in der Service-linked Rollenspalte nach den Diensten, für die Ja steht. Wählen Sie über einen Link Ja aus, um die Dokumentation zu einer serviceverknüpften Rolle für diesen Service anzuzeigen.

Service-linked Rollenberechtigungen für AWS STK.

AWS PCS verwendet die serviceverknüpfte Rolle mit dem Namen AWSServiceRoleForPCS— Erteilt AWS PCS die Erlaubnis, Amazon EC2 EC2-Ressourcen zu verwalten.

Die AWSServiceRoleForPCS serviceverknüpfte Rolle vertraut darauf, dass die folgenden Dienste die Rolle übernehmen:

- `pcs.amazonaws.com`

Die genannte Rollenberechtigungsrichtlinie AWSPCSServiceRolePolicyermöglicht es AWS PCS, Aktionen für bestimmte Ressourcen durchzuführen.

Sie müssen Berechtigungen konfigurieren, damit eine Benutzer, Gruppen oder Rollen eine serviceverknüpfte Rolle erstellen, bearbeiten oder löschen können. Weitere Informationen finden Sie unter Service-linked Rollenberechtigungen im IAM-Benutzerhandbuch.

Erstellen einer serviceverknüpften Rolle für AWS STK.

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. AWS PCS erstellt für Sie eine dienstverknüpfte Rolle, wenn Sie einen Cluster erstellen.

Bearbeiten einer serviceverknüpften Rolle für AWS STK.

AWS PCS erlaubt es Ihnen nicht, die AWSServiceRoleForPCS serviceverknüpfte Rolle zu bearbeiten. Da möglicherweise verschiedene Entitäten auf die Rolle verweisen, kann der Rollenname nach dem Erstellen einer serviceverknüpften Rolle nicht mehr geändert werden. Sie können jedoch die Beschreibung der Rolle mit IAM bearbeiten. Weitere Informationen finden Sie unter Bearbeiten einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Löschen einer serviceverknüpften Rolle für AWS STK.

Wenn Sie ein Feature oder einen Dienst, die bzw. der eine serviceverknüpften Rolle erfordert, nicht mehr benötigen, sollten Sie diese Rolle löschen. Auf diese Weise haben Sie keine ungenutzte juristische Stelle, die nicht aktiv überwacht oder verwaltet wird. Sie müssen jedoch die Ressourcen für Ihre serviceverknüpften Rolle zunächst bereinigen, bevor Sie sie manuell löschen können.

> **Note**
>
> Wenn der AWS PCS-Dienst die Rolle verwendet, wenn Sie versuchen, die Ressourcen zu löschen, schlägt das Löschen möglicherweise fehl. Wenn dies passiert, warten Sie einige Minuten und versuchen Sie es erneut.

Um AWS PCS-Ressourcen zu entfernen, die von AWSServiceRoleForPCS

Sie müssen alle Ihre Cluster löschen, um die AWSServiceRoleForPCS dienstverknüpfte Rolle zu löschen. Weitere Informationen finden Sie unter Löschen eines Clusters.

So löschen Sie die serviceverknüpfte Rolle mit IAM

Verwenden Sie die IAM-Konsole, die oder die AWS API AWS CLI, um die AWSServiceRoleForPCS serviceverknüpfte Rolle zu löschen. Weitere Informationen finden Sie unter Löschen einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Unterstützte Regionen für AWS Mit dem PCS-Dienst verknüpfte Rollen

AWS PCS unterstützt die Verwendung von serviceverknüpften Rollen in allen Regionen, in denen der Service verfügbar ist. Weitere Informationen finden Sie unter AWS -Regionen und Endpunkte.

Amazon EC2 Spot-Rolle für AWS STK.

Wenn Sie eine AWS PCS-Compute-Knotengruppe erstellen möchten, die Spot als Kaufoption verwendet, müssen Sie auch die AWSServiceRoleForEC2Spotserviceverknüpfte Rolle in Ihrer haben. AWS-Konto Sie können den folgenden AWS CLI Befehl verwenden, um die Rolle zu erstellen. Weitere Informationen finden Sie im AWS Identity and Access Management Benutzerhandbuch unter Erstellen einer dienstbezogenen Rolle und Erstellen einer Rolle zum Delegieren von Berechtigungen für einen AWS Dienst.

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

> **Note**
>
> Sie erhalten die folgende Fehlermeldung, wenn Sie AWS-Konto bereits über eine `AWSServiceRoleForEC2Spot` IAM-Rolle verfügen.
>
> ```
> An error occurred (InvalidInput) when calling the CreateServiceLinkedRole
> operation: Service role name AWSServiceRoleForEC2Spot has been taken in this
> account, please try a different suffix.
> ```

Mindestberechtigungen für AWS STK.

In diesem Abschnitt werden die IAM-Mindestberechtigungen beschrieben, die für eine IAM-Identität (Benutzer, Gruppe oder Rolle) zur Nutzung des Dienstes erforderlich sind.

Inhalt

- Mindestberechtigungen zur Verwendung von Capacity Blocks
- Mindestberechtigungen für einen Dienstadministrator

Mindestberechtigungen zur Verwendung von API-Aktionen

API-Aktion	Mindestberechtigungen	Zusätzliche Berechtigungen für die Konsole
CreateCluster	`ec2:CreateNetworkInterface, ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:GetSecurityGroupsForVpc, iam:CreateServiceLinkedRole, secretsmanager:CreateSecret, secretsmanager:TagResource, secretsmanager:RotateSecret, pcs:CreateCluster`	
ListClusters	`pcs:ListClusters`	
GetCluster	`pcs:GetCluster`	`ec2:DescribeSubnets`
DeleteCluster	`pcs:DeleteCluster`	
CreateComputeNodeGroup	`ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups,`	`iam:ListInstanceProfiles, ec2:DescribeImages, pcs:GetCluster`

API-Aktion	Mindestberechtigungen	Zusätzliche Berechtigungen für die Konsole
	`ec2:DescribeLaunchTemplates`, `ec2:DescribeLaunchTemplateVersions`, `ec2:DescribeInstanceTypes`, `ec2:DescribeInstanceTypeOfferings`, `ec2:RunInstances`, `ec2:CreateFleet`, `ec2:CreateTags`, `iam:PassRole`, `iam:GetInstanceProfile`, `pcs:CreateComputeNodeGroup`	
ListComputerNodeGroups	`pcs:ListComputeNodeGroups`	`pcs:GetCluster`
GetComputeNodeGroup	`pcs:GetComputeNodeGroup`	`ec2:DescribeSubnets`

API-Aktion	Mindestberechtigungen	Zusätzliche Berechtigungen für die Konsole
UpdateComputeNodeGroup	`ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:DescribeLaunchTemplates, ec2:DescribeLaunchTemplateVersions, ec2:DescribeInstanceTypes, ec2:DescribeInstanceTypeOfferings, ec2:RunInstances, ec2:CreateFleet, ec2:CreateTags, iam:PassRole, iam:GetInstanceProfile, pcs:UpdateComputeNodeGroup`	`pcs:GetComputeNodeGroup, iam:ListInstanceProfiles, ec2:DescribeImages, pcs:GetCluster`
DeleteComputeNodeGroup	`pcs:DeleteComputeNodeGroup`	
CreateQueue	`pcs:CreateQueue`	`pcs:ListComputeNodeGroups, pcs:GetCluster`
ListQueues	`pcs:ListQueues`	`pcs:GetCluster`
GetQueue	`pcs:GetQueue`	

API-Aktion	Mindestberechtigungen	Zusätzliche Berechtigungen für die Konsole
UpdateQueue	`pcs:UpdateQueue`	`pcs:ListComputeNodeGroups, pcs:GetQueue`
DeleteQueue	`pcs:DeleteQueue`	

Mindestberechtigungen zur Verwendung von Tags

Die folgenden Berechtigungen sind erforderlich, um Tags mit Ihren Ressourcen in AWS PCS zu verwenden.

```
pcs:ListTagsForResource,
pcs:TagResource,
pcs:UntagResource
```

Mindestberechtigungen zur Unterstützung von Protokollen

AWS PCS sendet Protokolldaten an Amazon CloudWatch Logs (CloudWatch Logs). Sie müssen sicherstellen, dass Ihre Identität über die Mindestberechtigungen zur Verwendung von CloudWatch Logs verfügt. Weitere Informationen finden Sie unter Überblick über die Verwaltung von Zugriffsberechtigungen für Ihre CloudWatch Logs-Ressourcen im Amazon CloudWatch Logs-Benutzerhandbuch.

Informationen zu den Berechtigungen, die für einen Service zum Senden von Protokollen an CloudWatch Logs erforderlich sind, finden Sie unter Aktivieren der Protokollierung von AWS Diensten im Amazon CloudWatch Logs-Benutzerhandbuch.

Mindestberechtigungen zur Verwendung von Capacity Blocks

Amazon EC2 Capacity Blocks for ML ist eine Amazon EC2 EC2-Kaufoption, mit der Sie im Voraus bezahlen können, um GPU-based Accelerated Computing-Instances innerhalb eines bestimmten Datums und Zeitbereichs zu reservieren, um Workloads mit kurzer Dauer zu unterstützen. Weitere Informationen finden Sie unter Verwenden von Amazon EC2 EC2-Kapazitätsblöcken für ML mit AWS PCS.

Sie entscheiden sich dafür, Capacity Blocks zu verwenden, wenn Sie eine Rechenknotengruppe erstellen oder aktualisieren. Die IAM-Identität, die Sie zum Erstellen oder Aktualisieren der Compute-Knotengruppe verwenden, muss über die folgenden Berechtigungen verfügen:

```
ec2:DescribeCapacityReservations,
ec2:DescribeCapacityBlocks,
ec2:DescribeCapacityBlockStatus
```

Mindestberechtigungen für einen Dienstadministrator

Die folgende IAM-Richtlinie legt die Mindestberechtigungen fest, die für eine IAM-Identität (Benutzer, Gruppe oder Rolle) erforderlich sind, um den AWS PCS-Service zu konfigurieren und zu verwalten.

> **Note**
>
> Benutzer, die den Dienst nicht konfigurieren und verwalten, benötigen diese Berechtigungen nicht. Benutzer, die nur Jobs ausführen, verwenden Secure Shell (SSH), um eine Verbindung zum Cluster herzustellen. AWS Identity and Access Management (IAM) kümmert sich nicht um die Authentifizierung oder Autorisierung für SSH.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PCSAccess",
      "Effect": "Allow",
      "Action": [
        "pcs:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "EC2Access",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface",
        "ec2:DescribeImages",
        "ec2:GetSecurityGroupsForVpc",
        "ec2:DescribeSubnets",
        "ec2:DescribeSecurityGroups",
```

```
        "ec2:DescribeVpcs",
        "ec2:DescribeLaunchTemplates",
        "ec2:DescribeLaunchTemplateVersions",
        "ec2:DescribeInstanceTypes",
        "ec2:DescribeInstanceTypeOfferings",
        "ec2:RunInstances",
        "ec2:CreateFleet",
        "ec2:CreateTags",
        "ec2:DescribeCapacityReservations",
        "ec2:DescribeCapacityBlocks",
        "ec2:DescribeCapacityBlockStatus"
      ],
      "Resource": "*"
    },
    {
      "Sid": "IamInstanceProfile",
      "Effect": "Allow",
      "Action": [
        "iam:GetInstanceProfile"
      ],
      "Resource": "*"
    },
    {
      "Sid": "IamPassRole",
      "Effect": "Allow",
      "Action": [
        "iam:PassRole"
      ],
      "Resource": [
        "arn:aws:iam::*:role/*/AWSPCS*",
        "arn:aws:iam::*:role/AWSPCS*",
        "arn:aws:iam::*:role/aws-pcs/*",
        "arn:aws:iam::*:role/*/aws-pcs/*"
      ],
      "Condition": {
        "StringEquals": {
           "iam:PassedToService": [
             "ec2.amazonaws.com"
           ]
        }
      }
    },
    {
      "Sid": "SLRAccess",
```

```
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": [
        "arn:aws:iam::*:role/aws-service-role/pcs.amazonaws.com/AWSServiceRoleFor*",
        "arn:aws:iam::*:role/aws-service-role/spot.amazonaws.com/AWSServiceRoleFor*"
      ],
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": [
            "pcs.amazonaws.com",
            "spot.amazonaws.com"
          ]
        }
      }
    },
    {
      "Sid": "AccessKMSKey",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:GenerateDataKey",
        "kms:CreateGrant",
        "kms:DescribeKey"
      ],
      "Resource": "*"
    },
    {
      "Sid": "SecretManagementAccess",
      "Effect": "Allow",
      "Action": [
        "secretsmanager:CreateSecret",
        "secretsmanager:TagResource",
        "secretsmanager:UpdateSecret",
        "secretsmanager:RotateSecret"
      ],
      "Resource": "*"
    },
    {
       "Sid": "ServiceLogsDelivery",
       "Effect": "Allow",
       "Action": [
```

```
          "pcs:AllowVendedLogDeliveryForResource",
          "logs:PutDeliverySource",
          "logs:PutDeliveryDestination",
          "logs:CreateDelivery"
        ],
        "Resource": "*"
    }
  ]
}
```

IAM-Instanzprofile für AWS Dienst für parallele Datenverarbeitung

Anwendungen, die auf einer EC2-Instance ausgeführt werden, müssen in allen AWS API-Anfragen, die sie stellen, AWS Anmeldeinformationen enthalten. Wir empfehlen Ihnen, eine IAM-Rolle zu verwenden, um temporäre Anmeldeinformationen auf der EC2-Instance zu verwalten. Sie können dafür ein Instance-Profil definieren und es an Ihre Instances anhängen. Weitere Informationen finden Sie unter IAM-Rollen für Amazon EC2 im Amazon Elastic Compute Cloud-Benutzerhandbuch.

> **Note**
>
> Wenn Sie die verwenden AWS-Managementkonsole , um eine IAM-Rolle für Amazon EC2 zu erstellen, erstellt die Konsole automatisch ein Instance-Profil und weist diesem den gleichen Namen wie die IAM-Rolle zu. Wenn Sie die AWS CLI IAM-Rolle mithilfe von AWS API-Aktionen oder einem AWS SDK erstellen, erstellen Sie das Instance-Profil als separate Aktion. Weitere Informationen finden Sie unter Instanzprofile im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Sie müssen den Amazon-Ressourcennamen (ARN) eines Instance-Profils angeben, wenn Sie Compute-Knotengruppen erstellen. Sie können verschiedene Instance-Profile für einige oder alle Compute-Knotengruppen wählen.

Voraussetzungen

IAM-Rolle des Instanzprofils

Die dem Instanzprofil zugeordnete IAM-Rolle muss `/aws-pcs/` in ihrem Pfad enthalten sein, oder ihr Name muss mit beginnen. `AWSPCS`

Beispiel für ARNs für eine IAM-Rolle

- `arn:aws:iam::*:role/AWSPCS-example-role-1`
- `arn:aws:iam::*:role/aws-pcs/example-role-2`

Berechtigungen

Die dem Instanzprofil für AWS PCS zugeordnete IAM-Rolle muss die folgende Richtlinie enthalten.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "pcs:RegisterComputeNodeGroupInstance"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

Zusätzliche Richtlinien

Erwägen Sie, verwaltete Richtlinien zum Instanzprofil hinzuzufügen. Beispiel:

- AmazonS3ReadOnlyAccessbietet schreibgeschützten Zugriff auf alle S3-Buckets.
- AmazonSSMManagedInstanceCoreaktiviert die Kernfunktionen des AWS Systems Manager Manager-Service, z. B. den Fernzugriff direkt von der Amazon Management Console aus.
- CloudWatchAgentServerPolicyenthält Berechtigungen, die für die Verwendung AmazonCloudWatchAgent auf Servern erforderlich sind.

Sie können auch Ihre eigenen IAM-Richtlinien angeben, die Ihren speziellen Anwendungsfall unterstützen.

Erstellen Sie ein Instanzprofil für AWS STK.

AWS PCS console

Wählen Sie „Basisprofil erstellen“, wenn Sie eine Compute-Knotengruppe erstellen, damit AWS PCS für Sie ein Profil mit der erforderlichen Mindestrichtlinie erstellt.

Amazon EC2 console

Sie können ein Instance-Profil direkt von der Amazon EC2 EC2-Konsole aus erstellen. Weitere Informationen finden Sie unter Verwenden von Instance-Profilen im AWS Identity and Access Management Benutzerhandbuch.

> **Important**
>
> Stellen Sie sicher, dass Sie das erforderliche Präfix `AWSPCS` im IAM-Rollennamen verwenden.

AWS CLI

Einrichtung eines Basisinstanzprofils mit AWS CLI

> **Note**
>
> Ersetzen Sie *`example-role`* in den folgenden Beispielen den Namen Ihrer IAM-Rolle.

1. Erstellen Sie eine IAM-Rolle mit dem `/aws-pcs/` Pfadattribut oder einem Namen, der mit beginnt. `AWSPCS`

   a. Kopieren Sie den folgenden Inhalt und fügen Sie ihn in eine neue Textdatei mit dem Namen `trust_policy.json` ein.

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

```
                "Service": [
                    "ec2.amazonaws.com"
                ]
            },
            "Action": [
                "sts:AssumeRole"
            ]
        }
    ]
}
```

b. Verwenden Sie einen der folgenden Befehle, um die IAM-Rolle zu erstellen.

```
aws iam create-role --path /aws-pcs/ --role-name example-role --assume-role-policy-document file://trust_policy.json
```

oder

```
aws iam create-role --role-name AWSPCS-example-role --assume-role-policy-document file://trust_policy.json
```

2. Fügen Sie Berechtigungen hinzu.

a. Kopieren Sie den folgenden Inhalt und fügen Sie ihn in eine neue Textdatei mit dem Namen ein`policy_document.json`.

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Action": [
                "pcs:RegisterComputeNodeGroupInstance"
            ],
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

b. Hängen Sie das Richtliniendokument an die Rolle an. Mit diesem Befehl wird die Richtlinie als Inline-Richtlinie angehängt.

```
aws iam put-role-policy \
    --role-name example-role \
    --policy-name pcsRegisterInstancePolicy \
    --policy-document file://policy_document.json
```

3. Erstellen Sie ein Instanzprofil. ***example-profile***Ersetzen Sie es durch den Namen Ihres Instanzprofils.

```
aws iam create-instance-profile --instance-profile-name example-profile
```

4. Ordnen Sie die IAM-Rolle dem Instanzprofil zu.

```
aws iam add-role-to-instance-profile \
   --instance-profile-name example-profile \
   --role-name example-role
```

Finden Sie Instanzprofile, die verwendet werden mit AWS STK.

1. Wenn Sie die genauen Namen Ihrer IAM-Rollen für AWS PCS nicht kennen, verwenden Sie den folgenden AWS CLI Befehl, um die IAM-Rollen aufzulisten, die die AWS PCS-Namensanforderungen erfüllen.

```
aws iam list-roles --query "Roles[?starts_with(RoleName, 'AWSPCS') ||
 contains(Path, '/aws-pcs/')].[RoleName]" --output text
```

2. Verwenden Sie den folgenden AWS CLI Befehl, um die Instanzprofile aufzulisten, die einer bestimmten IAM-Rolle zugeordnet sind. *role-name*Ersetzen Sie es durch den Namen einer IAM-Rolle, die die Anforderungen an den AWS PCS-Namen erfüllt.

```
aws iam list-instance-profiles-for-role --role-name role-name
```

Fehlerbehebung AWS Identität und Zugriff auf den Parallel Computing Service

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit AWS PCS und IAM auftreten können.

Themen

Ich bin nicht berechtigt, eine Aktion durchzuführen in AWS STK.

Wenn Sie eine Fehlermeldung erhalten, dass Sie nicht zur Durchführung einer Aktion berechtigt sind, müssen Ihre Richtlinien aktualisiert werden, damit Sie die Aktion durchführen können.

Der folgende Beispielfehler tritt auf, wenn der IAM-Benutzer `mateojackson` versucht, über die Konsole Details zu einer fiktiven *`my-example-widget`*-Ressource anzuzeigen, jedoch nicht über `pcs:`*`GetWidget`*-Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 pcs:GetWidget on resource: my-example-widget
```

In diesem Fall muss die Richtlinie für den Benutzer `mateojackson` aktualisiert werden, damit er mit der `pcs:`*`GetWidget`*-Aktion auf die *`my-example-widget`*-Ressource zugreifen kann.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich bin nicht berechtigt, iam auszuführen: PassRole

Wenn Sie die Fehlermeldung erhalten, dass Sie nicht autorisiert sind, die `iam:PassRole` Aktion auszuführen, müssen Ihre Richtlinien aktualisiert werden, damit Sie eine Rolle an AWS PCS übergeben können.

Einige AWS-Services ermöglichen es Ihnen, eine bestehende Rolle an diesen Dienst zu übergeben, anstatt eine neue Servicerolle oder eine dienstverknüpfte Rolle zu erstellen. Hierzu benötigen Sie Berechtigungen für die Übergabe der Rolle an den Dienst.

Der folgende Beispielfehler tritt auf, wenn ein IAM-Benutzer mit dem Namen `marymajor` versucht, die Konsole zu verwenden, um eine Aktion in AWS PCS auszuführen. Die Aktion erfordert jedoch, dass der Service über Berechtigungen verfügt, die durch eine Servicerolle gewährt werden. Mary besitzt keine Berechtigungen für die Übergabe der Rolle an den Dienst.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In diesem Fall müssen die Richtlinien von Mary aktualisiert werden, um die Aktion `iam:PassRole` ausführen zu können.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich möchte Personen außerhalb meiner Umgebung zulassen AWS-Konto um auf meine zuzugreifen AWS PCS-Ressourcen

Sie können eine Rolle erstellen, mit der Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation auf Ihre Ressourcen zugreifen können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Im Fall von Diensten, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (Access Control Lists, ACLs) verwenden, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob AWS PCS diese Funktionen unterstützt, finden Sie unterWie AWS Parallel Computing Service funktioniert mit IAM.
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für einen IAM-Benutzer in einem anderen AWS-Konto , den Sie besitzen.
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie AWS-Konten im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für Dritte.
- Informationen dazu, wie Sie über einen Identitätsverbund Zugriff gewähren, finden Sie unter Gewähren von Zugriff für extern authentifizierte Benutzer (Identitätsverbund) im IAM-Benutzerhandbuch.
- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Konformitätsprüfung für AWS Dienst für parallele Datenverarbeitung

Informationen darüber, ob AWS-Service ein AWS-Services in den Geltungsbereich bestimmter Compliance-Programme fällt, finden Sie unter Umfang nach Compliance-Programm AWS-Services unter . Wählen Sie dort das Compliance-Programm aus, an dem Sie interessiert sind. Allgemeine Informationen finden Sie unter AWS Compliance-Programme AWS .

Sie können Prüfberichte von Drittanbietern unter herunterladen AWS Artifact. Weitere Informationen finden Sie unter Berichte herunterladen unter .

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services hängt von der Vertraulichkeit Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. Weitere Informationen zu Ihrer Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services finden Sie in der AWS Sicherheitsdokumentation.

Resilienz in AWS Dienst für parallele Datenverarbeitung

Die AWS globale Infrastruktur basiert auf Availability AWS-Regionen Zones. AWS-Regionen bieten mehrere physisch getrennte und isolierte Availability Zones, die über Netzwerke mit niedriger Latenz, hohem Durchsatz und hoher Redundanz miteinander verbunden sind. Mithilfe von Availability Zones können Sie Anwendungen und Datenbanken erstellen und ausführen, die automatisch Failover zwischen Zonen ausführen, ohne dass es zu Unterbrechungen kommt. Availability Zones sind besser verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Weitere Informationen zu Availability Zones AWS-Regionen und Availability Zones finden Sie unter AWS Globale Infrastruktur.

Sicherheit der Infrastruktur in AWS Dienst für parallele Datenverarbeitung

Als verwalteter Dienst ist AWS Parallel Computing Service durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS Sicherheitsdiensten und zum AWS Schutz der Infrastruktur finden Sie unter AWS Cloud-Sicherheit. Informationen zum Entwerfen Ihrer AWS Umgebung unter Verwendung der bewährten Methoden für die Infrastruktursicherheit finden Sie unter Infrastructure Protection in Security Pillar AWS Well-Architected Framework.

Sie verwenden AWS veröffentlichte API-Aufrufe, um über das Netzwerk auf AWS PCS zuzugreifen. Kunden müssen Folgendes unterstützen:

- Transport Layer Security (TLS). Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Cipher-Suites mit Perfect Forward Secrecy (PFS) wie DHE (Ephemeral) oder ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Diffie-Hellman Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi.

Wenn AWS PCS einen Cluster erstellt, startet der Service den Slurm-Controller in einem diensteigenen Konto, getrennt von den Rechenknoten in Ihrem Konto. Um die Kommunikation zwischen dem Controller und den Rechenknoten zu überbrücken, erstellt AWS PCS ein kontenübergreifendes Elastic Network Interface (ENI) in Ihrer VPC. Der Slurm-Controller verwendet das ENI, um die Rechenknoten auf verschiedenen Ebenen zu verwalten und mit ihnen zu kommunizieren AWS-Konten, wodurch die Sicherheit und Isolierung der Ressourcen gewährleistet und gleichzeitig effiziente HPC- und Betriebsabläufe ermöglicht werden. AI/ML

Analyse und Management von Sicherheitslücken in AWS Dienst für parallele Datenverarbeitung

Konfiguration und IT-Kontrollen liegen in der gemeinsamen Verantwortung von Ihnen AWS und Ihnen. Weitere Informationen finden Sie im Modell der AWS gemeinsamen Verantwortung. AWS erledigt grundlegende Sicherheitsaufgaben für die dem Dienstkonto zugrunde liegende Infrastruktur, wie z. B. das Patchen des Betriebssystems auf Controller-Instanzen, die Firewallkonfiguration und die Notfallwiederherstellung der AWS Infrastruktur. Diese Verfahren wurden von qualifizierten Dritten überprüft und zertifiziert. Weitere Informationen finden Sie unter Bewährte Methoden für Sicherheit, Identität und Compliance.

> **Note**
>
> Slurm-Controller sind nicht verfügbar, solange wir sie aktualisieren. Laufende Jobs sind nicht betroffen. Jobs, die gesendet werden, wenn der Controller des Clusters nicht verfügbar ist, werden zurückgehalten, bis der Controller verfügbar ist.

Sie sind verantwortlich für die Sicherheit der zugrunde liegenden Infrastruktur in Ihrem AWS-Konto:

- Pflegen Sie Ihren Code, einschließlich Updates und Sicherheitspatches.

- Patchen und aktualisieren Sie das Betriebssystem im Amazon Machine Image (AMI) für Ihre Rechenknotengruppen und aktualisieren Sie Ihre Rechenknotengruppen, um das aktualisierte AMI zu verwenden.
- Aktualisieren Sie den Scheduler, um die unterstützten Versionen beizubehalten. Aktualisieren Sie das AMI für Ihre Compute-Knotengruppen und aktualisieren Sie Ihre Compute-Knotengruppe, um das aktualisierte AMI zu verwenden.
- Authentifizieren und verschlüsseln Sie die Kommunikation zwischen Benutzerclients und den Knoten, mit denen sie sich verbinden.

Weitere Informationen zur Aktualisierung des AMI für Ihre Compute-Knotengruppen finden Sie unterAmazon Machine Images (AMIs) für AWS STK..

Cross-service verwirrter Stellvertreter, Prävention

Das Confused-Deputy-Problem ist ein Sicherheitsproblem, bei dem eine Entität, die nicht über die Berechtigung zum Ausführen einer Aktion verfügt, eine Entität mit größeren Rechten zwingen kann, die Aktion auszuführen. Im AWS Fall eines dienststellenübergreifenden Identitätswechsels kann es zu einem Problem mit verwirrten Stellvertretern kommen. Cross-service Ein Identitätswechsel kann auftreten, wenn ein Dienst (der anrufende Dienst) einen anderen Dienst (den angerufenen Dienst) aufruft. Der aufrufende Service kann manipuliert werden, um seine Berechtigungen zu verwenden, um Aktionen auf die Ressourcen eines anderen Kunden auszuführen, für die er sonst keine Zugriffsberechtigung haben sollte. Um dies zu verhindern, bietet AWS Tools, mit denen Sie Ihre Daten für alle Services mit Serviceprinzipalen schützen können, die Zugriff auf Ressourcen in Ihrem Konto erhalten haben.

Wir empfehlen, die Kontextschlüssel `aws:SourceArn`und die `aws:SourceAccount`globalen Bedingungsschlüssel in Ressourcenrichtlinien zu verwenden, um die Berechtigungen einzuschränken, die AWS Parallel Computing Service (AWS PCS) der Ressource einem anderen Dienst erteilt. Verwenden Sie `aws:SourceArn`, wenn Sie nur eine Ressource mit dem betriebsübergreifenden Zugriff verknüpfen möchten. Verwenden Sie `aws:SourceAccount`, wenn Sie zulassen möchten, dass Ressourcen in diesem Konto mit der betriebsübergreifenden Verwendung verknüpft werden.

Der effektivste Weg, um sich vor dem Confused-Deputy-Problem zu schützen, ist die Verwendung des globalen Bedingungskontext-Schlüssels `aws:SourceArn` mit dem vollständigen ARN der Ressource. Wenn Sie den vollständigen ARN der Ressource nicht kennen oder wenn Sie

mehrere Ressourcen angeben, verwenden Sie den globalen Kontextbedingungsschlüssel `aws:SourceArn` mit Platzhalterzeichen (*) für die unbekannten Teile des ARN. Beispiel, `arn:aws:servicename:*:123456789012:*`.

Wenn der `aws:SourceArn`-Wert die Konto-ID nicht enthält, z. B. einen Amazon-S3-Bucket-ARN, müssen Sie beide globale Bedingungskontextschlüssel verwenden, um Berechtigungen einzuschränken.

Der Wert von `aws:SourceArn` muss ein Cluster-ARN sein.

Das folgende Beispiel zeigt, wie Sie die Kontextschlüssel `aws:SourceArn` und die `aws:SourceAccount` globalen Bedingungsschlüssel in AWS PCS verwenden können, um das Problem des verwirrten Stellvertreters zu vermeiden.

```
{
"Version": "2012-10-17",
  "Statement": {
"Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "pcs.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": [
          "arn:aws:pcs:us-east-1:123456789012:cluster/*"
        ]
      },
      "StringEquals": {
        "aws:SourceAccount": "123456789012"
      }
    }
  }
}
```

IAM-Rolle für Amazon EC2 EC2-Instances, die als Teil einer Compute-Knotengruppe bereitgestellt werden

AWS PCS orchestriert automatisch die Amazon EC2 EC2-Kapazität für jede der konfigurierten Rechenknotengruppen in einem Cluster. Bei der Erstellung einer Rechenknotengruppe müssen

Benutzer über das Feld ein IAM-Instance-Profil angeben. `iamInstanceProfileArn` Das Instanzprofil gibt die Berechtigungen an, die den bereitgestellten EC2-Instances zugeordnet sind. AWS PCS akzeptiert jede Rolle, die ein Rollennamenpräfix oder `/aws-pcs/` Teil des Rollenpfads hat`AWSPCS`. Die `iam:PassRole` Berechtigung ist für die IAM-Identität (Benutzer oder Rolle) erforderlich, die eine Compute-Knotengruppe erstellt oder aktualisiert. Wenn ein Benutzer die `CreateComputeNodeGroup` oder `UpdateComputeNodeGroup` API-Aktionen aufruft, prüft AWS PCS, ob der Benutzer die `iam:PassRole` Aktion ausführen darf.

Die folgende Beispielrichtlinie gewährt die Berechtigung, nur IAM-Rollen weiterzugeben, deren Name mit `AWSPCS` beginnt.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::123456789012:role/AWSPCS*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "ec2.amazonaws.com"
                    ]
                }
            }
        }
    ]
}
```

Bewährte Sicherheitsmethoden für AWS Dienst für parallele Datenverarbeitung

In diesem Abschnitt werden bewährte Sicherheitsmethoden beschrieben, die speziell für AWS Parallel Computing Service (AWS PCS) gelten. Weitere Informationen zu bewährten Sicherheitsmethoden finden Sie unter Bewährte Methoden für Sicherheit, Identität und Compliance. AWS

AMI-related Sicherheit

- Verwenden Sie keine AWS PCS-Beispiel-AMIs für Produktionsworkloads. Die Beispiel-AMIs werden nicht unterstützt und sind nur für Tests vorgesehen.
- Aktualisieren Sie regelmäßig das Betriebssystem und die Software im AMI für Ihre Compute-Knotengruppen, um Sicherheitslücken zu minimieren.
- Verwenden Sie nur authentifizierte offizielle AWS PCS-Pakete, die von offiziellen AWS Quellen heruntergeladen wurden.
- Aktualisieren Sie regelmäßig die AWS PCS-Pakete im AMI für Compute-Knotengruppen und aktualisieren Sie die Compute-Knoten so, dass sie das aktualisierte AMI verwenden. Erwägen Sie, diesen Prozess zu automatisieren, um Sicherheitslücken zu minimieren.

Weitere Informationen finden Sie unter Benutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS.

Sicherheit von Slurm Workload Manager

- Implementieren Sie Zugriffskontrollen und Netzwerkeinschränkungen, um die Slurm-Kontroll- und Rechenknoten zu sichern. Erlauben Sie nur vertrauenswürdigen Benutzern und Systemen, Jobs einzureichen und auf Slurm-Verwaltungsbefehle zuzugreifen.
- Verwenden Sie die integrierten Sicherheitsfunktionen von Slurm, wie z. B. die Slurm-Authentifizierung, um sicherzustellen, dass Job-Eingaben und Kommunikation authentifiziert werden.
- Aktualisieren Sie die Slurm-Versionen, um einen reibungslosen Betrieb und die Cluster-Unterstützung aufrechtzuerhalten.

> **Important**
>
> Jeder Cluster, der eine Version von Slurm verwendet, die das Ende der Support-Laufzeit (EOSL) erreicht hat, wird sofort gestoppt. Verwenden Sie den Link oben auf den Seiten mit den Benutzerhandbüchern, um den RSS-Feed für die AWS PCS-Dokumentation zu abonnieren und eine Benachrichtigung zu erhalten, wenn sich eine Slurm-Version EOSL nähert.

Weitere Informationen finden Sie unter Slurm-Versionen in AWS STK..

- Wechseln Sie regelmäßig die Cluster-Geheimnisse, um die Einhaltung der Sicherheitsbestimmungen zu gewährleisten und potenzielle Sicherheitslücken zu beheben. Dies ist für die Einhaltung von HIPAA und FedRAMP erforderlich.

  Weitere Informationen finden Sie unter Rotierende Clustergeheimnisse in AWS PCS.

Überwachung und Protokollierung

- Verwenden Sie Amazon CloudWatch Logs und AWS CloudTrail , um Aktionen in Ihren Clustern zu überwachen und aufzuzeichnen und AWS-Konto. Verwenden Sie die Daten zur Fehlerbehebung und Prüfung.

Netzwerksicherheit

- Stellen Sie Ihre AWS PCS-Cluster in einer separaten VPC bereit, um Ihre HPC-Umgebung von anderem Netzwerkverkehr zu isolieren.
- Verwenden Sie Sicherheitsgruppen und Network Access Control Lists (ACLs), um den ein- und ausgehenden Datenverkehr zu PCS-Instanzen und Subnetzen zu AWS kontrollieren.
- Verwenden Sie AWS PrivateLink VPC VPC-Endpunkte, um den Netzwerkverkehr zwischen Ihren Clustern und anderen AWS Diensten innerhalb des AWS Netzwerks aufrechtzuerhalten. Weitere Informationen finden Sie unter Zugriff AWS Parallel Computing Service mithilfe eines Schnittstellenendpunkts (AWS PrivateLink).

Protokollierung und Überwachung für AWS PCS

Die Überwachung ist ein wichtiger Bestandteil der Aufrechterhaltung der Zuverlässigkeit, Verfügbarkeit und Leistung von AWS PCS und Ihren anderen AWS-Ressourcen. AWS bietet die folgenden Überwachungstools, um AWS PCS zu überwachen, zu melden, wenn etwas nicht stimmt, und gegebenenfalls automatische Maßnahmen zu ergreifen:

- Amazon CloudWatch überwacht Ihre AWS Ressourcen und die Anwendungen, auf denen Sie laufen, AWS in Echtzeit. Sie können Kennzahlen erfassen und verfolgen, benutzerdefinierte Dashboards erstellen und Alarme festlegen, die Sie benachrichtigen oder Maßnahmen ergreifen, wenn eine bestimmte Metrik einen von Ihnen festgelegten Schwellenwert erreicht. Sie können beispielsweise die CPU-Auslastung oder andere Kennzahlen Ihrer Amazon EC2 EC2-Instances CloudWatch verfolgen und bei Bedarf automatisch neue Instances starten. Weitere Informationen finden Sie im CloudWatch Amazon-Benutzerhandbuch.
- Mit Amazon CloudWatch Logs können Sie Ihre Protokolldateien von Amazon EC2 EC2-Instances und anderen Quellen überwachen CloudTrail, speichern und darauf zugreifen. CloudWatch Logs können Informationen in den Protokolldateien überwachen und Sie benachrichtigen, wenn bestimmte Schwellenwerte erreicht werden. Sie können Ihre Protokolldaten auch in einem sehr robusten Speicher archivieren. Weitere Informationen finden Sie im Amazon CloudWatch Logs-Benutzerhandbuch.
- AWS CloudTrailerfasst API-Aufrufe und zugehörige Ereignisse, die von oder im Namen Ihres AWS Kontos getätigt wurden, und übermittelt die Protokolldateien an einen von Ihnen angegebenen Amazon S3 S3-Bucket. Sie können feststellen, welche Benutzer und Konten angerufen wurden AWS, von welcher Quell-IP-Adresse aus die Anrufe getätigt wurden und wann die Aufrufe erfolgten. Weitere Informationen finden Sie im AWS CloudTrail -Benutzerhandbuch.

Auftragsabschlussprotokolle in AWS PCS

Auftragsabschlussprotokolle enthalten wichtige Informationen zu Ihren AWS Parallel Computing Service (AWS PCS) -Jobs, sobald sie abgeschlossen sind, ohne dass zusätzliche Kosten anfallen. Sie können andere AWS Dienste verwenden, um auf Ihre Protokolldaten zuzugreifen und diese zu verarbeiten, z. B. Amazon CloudWatch Logs, Amazon Simple Storage Service (Amazon S3) und Amazon Data Firehose. AWS PCS zeichnet Metadaten zu Ihren Jobs auf, z. B. die folgenden.

- Job-ID und Name
- Benutzer- und Gruppeninformationen

- Jobstatus (z. B.`COMPLETED`,`FAILED`,`CANCELLED`)
- Verwendete Partition
- Zeitlimits
- Beginn, Ende, Einreichung und zulässige Zeiten
- Liste und Anzahl der Knoten
- Anzahl Prozessoren
- Arbeitsverzeichnis
- Ressourcennutzung (CPU, Arbeitsspeicher)
- Exit-Codes
- Knotendetails (Namen, Instanz IDs, Instanztypen)

Inhalt

Voraussetzungen

Der IAM-Prinzipal, der den AWS PCS-Cluster verwaltet, muss die `pcs:AllowVendedLogDeliveryForResource` Aktion zulassen.

Die folgende Beispiel-IAM-Richtlinie gewährt die erforderlichen Berechtigungen.

JSON

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

```
            "Effect": "Allow",
            "Action": ["pcs:AllowVendedLogDeliveryForResource"],
            "Resource": [
               "arn:aws:pcs:*::cluster/*"
            ]
         }
      ]
}
```

Richten Sie Protokolle zum Abschluss von Aufträgen ein

Sie können Auftragsabschlussprotokolle für Ihren AWS PCS-Cluster mit dem AWS-Managementkonsole oder einrichten AWS CLI.

AWS-Managementkonsole

So richten Sie Auftragsabschlussprotokolle mit der Konsole ein

1. Öffnen Sie die AWS PCS-Konsole.
2. Klicken Sie im Navigationsbereich auf Cluster.
3. Wählen Sie den Cluster aus, dem Sie Auftragsabschlussprotokolle hinzufügen möchten.
4. Wählen Sie auf der Seite mit den Cluster-Details die Registerkarte Logs aus.
5. Wählen Sie unter Job Completion Logs die Option Hinzufügen aus, um bis zu 3 Log-Lieferziele aus CloudWatch Logs, Amazon S3 und Firehose hinzuzufügen.
6. Wählen Sie Protokollzustellungen aktualisieren aus.

AWS CLI

Um Protokolle zum Abschluss von Jobs einzurichten mit dem AWS CLI

1. Erstellen Sie ein Ziel für die Protokollzustellung:

```
aws logs put-delivery-destination --region region \
  --name pcs-logs-destination \
  --delivery-destination-configuration \
  destinationResourceArn=resource-arn
```

Ersetzen Sie:

- *region*— Der AWS-Region Ort, an dem Sie das Ziel erstellen möchten, z. B. `us-east-1`
- *pcs-logs-destination*— Ein Name für das Ziel
- *resource-arn*— Der Amazon-Ressourcenname (ARN) einer CloudWatch Logs-Protokollgruppe, eines S3-Buckets oder eines Firehose-Lieferstreams.

Weitere Informationen finden Sie PutDeliveryDestinationin der Amazon CloudWatch Logs API-Referenz.

2. Legen Sie den PCS-Cluster als Quelle für die Protokollzustellung fest:

```
aws logs put-delivery-source --region region \
  --name cluster-logs-source-name \
  --resource-arn cluster-arn \
  --log-type PCS_JOBCOMP_LOGS
```

Ersetzen Sie:

- *region*— Die AWS-Region Ihres Clusters, wie `us-east-1`
- *cluster-logs-source-name*— Ein Name für die Quelle
- *cluster-arn*— der ARN Ihres AWS PCS-Clusters

Weitere Informationen finden Sie PutDeliverySourcein der Amazon CloudWatch Logs API-Referenz.

3. Connect die Lieferquelle mit dem Lieferziel:

```
aws logs create-delivery --region region \
  --delivery-source-name cluster-logs-source \
  --delivery-destination-arn destination-arn
```

Ersetzen Sie:

- *region*— Die AWS-Region, wie `us-east-1`
- *cluster-logs-source*— Der Name Ihrer Lieferquelle
- *destination-arn*— Die ARN Ihres Lieferziels

Weitere Informationen finden Sie CreateDeliveryin der Amazon CloudWatch Logs API-Referenz.

Wie finde ich die Protokolle zum Abschluss von Aufträgen

Sie können Protokollziele in CloudWatch Logs und Amazon S3 konfigurieren. AWS PCS verwendet die folgenden strukturierten Pfad- und Dateinamen.

CloudWatch Logs

AWS PCS verwendet das folgende Namensformat für den CloudWatch Logs-Stream:

```
AWSLogs/PCS/cluster-id/jobcomp.log
```

Beispiel: `AWSLogs/PCS/pcs_abc123de45/jobcomp.log`

Amazon S3

AWS PCS verwendet das folgende Namensformat für den S3-Pfad:

```
AWSLogs/account-id/PCS/region/cluster-id/jobcomp/year/month/day/hour/
```

Beispiel: `AWSLogs/111122223333/PCS/us-east-1/pcs_abc123de45/jobcomp/2025/06/19/11/`

AWS PCS verwendet das folgende Namensformat für die Protokolldateien:

```
PCS_jobcomp_year-month-day-hour_cluster-id_random-id.log.gz
```

Beispiel: `PCS_jobcomp_2025-06-19-11_pcs_abc123de45_04be080b.log.gz`

Protokollfelder für den Abschluss von Aufträgen

AWS PCS schreibt Protokolldaten zur Auftragsabwicklung als JSON-Objekte. Der JSON-Container `jobcomp` enthält Jobdetails. In der folgenden Tabelle werden die Felder im `jobcomp` Container beschrieben. Einige Felder sind nur unter bestimmten Umständen vorhanden, z. B. bei Array-Jobs oder heterogenen Jobs.

Protokollfelder für den Abschluss von Aufträgen

Name	Beispielwert	Erforderlich	Hinweise
`job_id`	`11`	Ja	Immer mit Wert präsent
`user`	`"root"`	Ja	Immer mit Wert präsent
`user_id`	`0`	Ja	Immer mit Wert präsent
`group`	`"root"`	Ja	Immer mit Wert präsent
`group_id`	`0`	Ja	Immer mit Wert präsent
`name`	`"wrap"`	Ja	Immer mit Wert präsent
`job_state`	`"COMPLETED"`	Ja	Immer mit Wert präsent
`partition`	`"Hydra-MpiQueue-abcdef01-7"`	Ja	Immer mit Wert präsent
`time_limit`	`"UNLIMITED"`	Ja	Immer präsent, könnte es aber sein `"UNLIMITED"`
`start_time`	`"2025-06-19T10:58:57"`	Ja	Immer präsent, könnte es aber sein `"Unknown"`
`end_time`	`"2025-06-19T10:58:57"`	Ja	Immer präsent, könnte es aber sein `"Unknown"`
`node_list`	`"Hydra-MpiNG-abcdef01-2345-1"`	Ja	Immer wertvoll präsent
`node_cnt`	`1`	Ja	Immer mit Wert präsent
`proc_cnt`	`1`	Ja	Immer mit Wert präsent

Name	Beispielwert	Erforderlich	Hinweise
work_dir	"/root"	Ja	Immer präsent, könnte es aber sein "Unknown"
reservation_name	"weekly_maintenance"	Ja	Immer vorhanden, könnte aber eine leere Zeichenfolge sein ""
tres.cpu	1	Ja	Immer mit Wert präsent
tres.mem.val	600	Ja	Immer mit Wert präsent
tres.mem.unit	"M"	Ja	Kann sein "M" oder "bb"
tres.node	1	Ja	Immer mit Wert präsent
tres.billing	1	Ja	Immer mit Wert präsent
account	"finance"	Ja	Immer vorhanden, kann aber eine leere Zeichenfolge sein ""
qos	"normal"	Ja	Immer vorhanden, könnte aber eine leere Zeichenfolge sein ""
wc_key	"project_1"	Ja	Immer vorhanden, könnte aber eine leere Zeichenfolge sein ""
cluster	"unknown"	Ja	Immer präsent, könnte es aber sein "unknown"
submit_time	"2025-06-19T10:55:46"	Ja	Immer präsent, könnte es aber sein "Unknown"

Name	Beispielwert	Erforderlich	Hinweise
`eligible_time`	`"2025-06-19T10:55:46"`	Ja	Immer präsent, könnte es aber sein `"Unknown"`
`array_job_id`	`12`	Nein	Nur vorhanden, wenn es sich bei dem Job um einen Array-Job handelt
`array_task_id`	`1`	Nein	Nur vorhanden, wenn es sich bei dem Job um einen Array-Job handelt
`het_job_id`	`10`	Nein	Nur vorhanden, wenn es sich bei dem Job um einen heterogenen Job handelt
`het_job_offset`	`0`	Nein	Nur vorhanden, wenn es sich bei der Tätigkeit um eine heterogene Tätigkeit handelt
`derived_exit_code_status`	`0`	Ja	Immer wertvoll präsent
`derived_exit_code_signal`	`0`	Ja	Immer mit Wert präsent
`exit_code_status`	`0`	Ja	Immer mit Wert präsent
`exit_code_signal`	`0`	Ja	Immer mit Wert präsent
`node_details[0].name`	`"Hydra-MpiNG-abcdef01-2345-1"`	Nein	Immer präsent, `node_details` könnte es aber sein `"[]"`

Name	Beispielwert	Erforderlich	Hinweise
`node_details[0].instance_id`	`"i-0abcdef01234567a"`	Nein	Immer präsent, `node_details` könnte es aber sein `"[]"`
`node_details[0].instance_type`	`"t4g.micro"`	Nein	Immer präsent, `node_details` könnte es aber sein `"[]"`

Beispiele für Protokolle zum Abschluss von Aufträgen

Die folgenden Beispiele zeigen Auftragsabschlussprotokolle für verschiedene Auftragstypen und -status:


```
{ "jobcomp": { "job_id": 1, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T16:32:57",
 "end_time": "2025-06-19T16:33:03", "node_list": "Hydra-MpiNG-abcdef01-2345-
[1-2]", "node_cnt": 2, "proc_cnt": 2, "work_dir": "/usr/bin", "reservation_name":
 "", "tres": { "cpu": 2, "mem": { "val": 1944, "unit": "M" }, "node": 2,
 "billing": 2 }, "account": "", "qos": "", "wc_key": "", "cluster": "unknown",
 "submit_time": "2025-06-19T16:29:40", "eligible_time": "2025-06-19T16:29:41",
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc123def45678", "instance_type": "t4g.micro" }, { "name":
 "Hydra-MpiNG-abcdef01-2345-2", "instance_id": "i-0def456abc78901", "instance_type":
 "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 2, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T16:33:13",
 "end_time": "2025-06-19T16:33:14", "node_list": "Hydra-MpiNG-abcdef01-2345-
[1-2]", "node_cnt": 2, "proc_cnt": 2, "work_dir": "/usr/bin", "reservation_name":
 "", "tres": { "cpu": 2, "mem": { "val": 1944, "unit": "M" }, "node": 2,
 "billing": 2 }, "account": "", "qos": "", "wc_key": "", "cluster": "unknown",
 "submit_time": "2025-06-19T16:33:13", "eligible_time": "2025-06-19T16:33:13",
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc123def45678", "instance_type": "t4g.micro" }, { "name":
```

```
 "Hydra-MpiNG-abcdef01-2345-2", "instance_id": "i-0def456abc78901", "instance_type":
 "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 3, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T22:58:57", "end_time":
 "2025-06-19T22:58:57", "node_list": "Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 972, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T22:55:46",
 "eligible_time": "2025-06-19T22:55:46", "derived_exit_code_status": 0,
 "derived_exit_code_signal": 0, "exit_code_status": 0, "exit_code_signal":
 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1", "instance_id":
 "i-0abc234def56789", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 4, "user": "root", "user_id": 0, "group": "root",
 "group_id": 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-
MpiQueue-abcdef01-7", "time_limit": "525600", "start_time": "2025-06-19T23:04:27",
 "end_time": "2025-06-19T23:04:27", "node_list": "Hydra-MpiNG-abcdef01-2345-
[1-2]", "node_cnt": 2, "proc_cnt": 2, "work_dir": "/root", "reservation_name":
 "", "tres": { "cpu": 2, "mem": { "val": 1944, "unit": "M" }, "node": 2,
 "billing": 2 }, "account": "", "qos": "", "wc_key": "", "cluster": "unknown",
 "submit_time": "2025-06-19T23:01:38", "eligible_time": "2025-06-19T23:01:38",
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc234def56789", "instance_type": "t4g.micro" }, { "name":
 "Hydra-MpiNG-abcdef01-2345-2", "instance_id": "i-0def345abc67890", "instance_type":
 "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 5, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "FAILED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:09:00", "end_time":
 "2025-06-19T23:09:00", "node_list": "(null)", "node_cnt": 0, "proc_cnt": 0,
 "work_dir": "/root", "reservation_name": "", "tres": { "cpu": 1, "mem": { "val":
 1, "unit": "G" }, "node": 1, "billing": 1 }, "account": "", "qos": "", "wc_key":
 "", "cluster": "unknown", "submit_time": "2025-06-19T23:09:00", "eligible_time":
 "2025-06-19T23:09:00", "derived_exit_code_status": 0, "derived_exit_code_signal": 0,
 "exit_code_status": 0, "exit_code_signal": 1, "node_details": [] } }
{ "jobcomp": { "job_id": 6, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "CANCELLED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:09:36",
 "end_time": "2025-06-19T23:09:36", "node_list": "(null)", "node_cnt": 0, "proc_cnt":
 0, "work_dir": "/root", "reservation_name": "", "tres": { "cpu": 1, "mem":
 { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "", "qos":
 "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:09:35",
 "eligible_time": "2025-06-19T23:09:36", "het_job_id": 6, "het_job_offset": 0,
```

```
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status": 0,
 "exit_code_signal": 1, "node_details": [] } }
{ "jobcomp": { "job_id": 7, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "CANCELLED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:10:03",
 "end_time": "2025-06-19T23:10:03", "node_list": "(null)", "node_cnt": 0, "proc_cnt":
 0, "work_dir": "/root", "reservation_name": "", "tres": { "cpu": 1, "mem":
 { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "", "qos":
 "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:10:03",
 "eligible_time": "2025-06-19T23:10:03", "het_job_id": 7, "het_job_offset": 0,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status": 0,
 "exit_code_signal": 1, "node_details": [] } }
{ "jobcomp": { "job_id": 8, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:11:24", "end_time":
 "2025-06-19T23:11:24", "node_list": "Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:11:23",
 "eligible_time": "2025-06-19T23:11:23", "het_job_id": 8, "het_job_offset": 0,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc234def56789", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 9, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:11:24", "end_time":
 "2025-06-19T23:11:24", "node_list": "Hydra-MpiNG-abcdef01-2345-2", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:11:23",
 "eligible_time": "2025-06-19T23:11:23", "het_job_id": 8, "het_job_offset": 1,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-2",
 "instance_id": "i-0def345abc67890", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 10, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:12:24", "end_time":
 "2025-06-19T23:12:24", "node_list":"Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:12:14",
 "eligible_time": "2025-06-19T23:12:14", "het_job_id": 10, "het_job_offset": 0,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
```

```
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc234def56789", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 11, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:12:24", "end_time":
 "2025-06-19T23:12:24", "node_list":"Hydra-MpiNG-abcdef01-2345-2", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 600, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:12:14",
 "eligible_time": "2025-06-19T23:12:14", "het_job_id": 10, "het_job_offset": 1,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-2",
 "instance_id": "i-0def345abc67890", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 13, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:47:57", "end_time":
 "2025-06-19T23:47:58", "node_list":"Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 972, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:43:56",
 "eligible_time": "2025-06-19T23:43:56" , "array_job_id": 12, "array_task_id": 1,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc345def67890", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 12, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:47:58", "end_time":
 "2025-06-19T23:47:58", "node_list":"Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 972, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:43:56",
 "eligible_time": "2025-06-19T23:43:56" , "array_job_id": 12, "array_task_id": 2,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc345def67890", "instance_type": "t4g.micro" } ] } }
```


Scheduler loggt sich in AWS PCS ein

Sie können AWS PCS so konfigurieren, dass detaillierte Protokolldaten von Ihrem Cluster-Scheduler an Amazon CloudWatch Logs, Amazon Simple Storage Service (Amazon S3) und Amazon Data Firehose gesendet werden. Dies kann bei der Überwachung und Fehlerbehebung hilfreich sein.

AWS PCS übermittelt Logs von den folgenden Slurm-Daemons über den `PCS_SCHEDULER_LOGS` Protokolltyp:

- **`slurmctld`**— Der Slurm-Controller-Daemon. Verfügbar für alle unterstützten Slurm-Versionen.
- **`slurmdbd`**— Der Slurm-Datenbank-Daemon. Verfügbar für Slurm 24.11 und höher.
- **`slurmrestd`**— Der Slurm-REST-API-Daemon. Verfügbar für Slurm 25.05 und höher.

Cluster, für die der `PCS_SCHEDULER_LOGS` Versand bereits konfiguriert ist, beginnen automatisch mit dem Empfang `slurmdbd` und `slurmrestd` protokollieren, wenn sie eine unterstützte Slurm-Version ausführen. Es ist keine zusätzliche Konfiguration erforderlich.

Inhalt

Voraussetzungen

Der IAM-Principal, der den AWS PCS-Cluster verwaltet, muss die `pcs:AllowVendedLogDeliveryForResource` Aktion zulassen.

Die folgende Beispiel-IAM-Richtlinie gewährt die erforderlichen Berechtigungen.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Sid": "PcsAllowVendedLogsDelivery",
         "Effect": "Allow",
         "Action": ["pcs:AllowVendedLogDeliveryForResource"],
         "Resource": [
            "arn:aws:pcs:*::cluster/*"
         ]
      }
```

```
    ]
}
```

Richten Sie Scheduler-Protokolle ein

Sie können Scheduler-Protokolle für Ihren AWS PCS-Cluster mit dem AWS-Managementkonsole oder einrichten. AWS CLI

AWS-Managementkonsole

Um Scheduler-Logs mit der Konsole einzurichten

1. Öffnen Sie die AWS PCS-Konsole.
2. Klicken Sie im Navigationsbereich auf Cluster.
3. Wählen Sie den Cluster aus, zu dem Sie Scheduler-Logs hinzufügen möchten.
4. Wählen Sie auf der Seite mit den Cluster-Details die Registerkarte Logs aus.
5. Wählen Sie unter Scheduler Logs die Option Hinzufügen aus, um bis zu 3 Log-Lieferziele aus CloudWatch Logs, Amazon S3 und Firehose hinzuzufügen.
6. Wählen Sie Protokollzustellungen aktualisieren aus.

AWS CLI

Um Scheduler-Logs einzurichten mit AWS CLI

1. Erstellen Sie ein Ziel für die Protokollzustellung:

```
aws logs put-delivery-destination --region region \
  --name pcs-logs-destination \
  --delivery-destination-configuration \
  destinationResourceArn=resource-arn
```

Ersetzen Sie:

- *region*— Der AWS-Region Ort, an dem Sie das Ziel erstellen möchten, z. B. `us-east-1`
- *pcs-logs-destination*— Ein Name für das Ziel
- *resource-arn*— Der Amazon-Ressourcenname (ARN) einer CloudWatch Logs-Protokollgruppe, eines S3-Buckets oder eines Firehose-Lieferstreams.

Weitere Informationen finden Sie PutDeliveryDestinationin der Amazon CloudWatch Logs API-Referenz.

2. Legen Sie den PCS-Cluster als Quelle für die Protokollzustellung fest:

```
aws logs put-delivery-source --region region \
  --name cluster-logs-source-name \
  --resource-arn cluster-arn \
  --log-type PCS_SCHEDULER_LOGS
```

Ersetzen Sie:

- *region*— Die AWS-Region Ihres Clusters, wie `us-east-1`
- *cluster-logs-source-name*— Ein Name für die Quelle
- *cluster-arn*— der ARN Ihres AWS PCS-Clusters

Weitere Informationen finden Sie PutDeliverySourcein der Amazon CloudWatch Logs API-Referenz.

3. Connect die Lieferquelle mit dem Lieferziel:

```
aws logs create-delivery --region region \
  --delivery-source-name cluster-logs-source \
  --delivery-destination-arn destination-arn
```

Ersetzen Sie:

- *region*— Die AWS-Region, wie `us-east-1`
- *cluster-logs-source*— Der Name Ihrer Lieferquelle
- *destination-arn*— Die ARN Ihres Lieferziels

Weitere Informationen finden Sie CreateDeliveryin der Amazon CloudWatch Logs API-Referenz.

Pfade und Namen von Log-Streams im Scheduler

Der Pfad und der Name der AWS PCS-Scheduler-Protokolle hängen vom Zieltyp ab.

Der `${log_name}` Wert in den folgenden Pfaden ist`slurmctld`, oder `slurmdbdslurmrestd`, abhängig vom Daemon, der das Protokoll erstellt hat.

- CloudWatch Protokolle
  - Ein CloudWatch Logs-Stream folgt dieser Namenskonvention.

    ```
    AWSLogs/PCS/${cluster_id}/${log_name}_${scheduler_major_version}.log
    ```

    Example

    ```
    AWSLogs/PCS/abcdef0123/slurmctld_25.11.log
    AWSLogs/PCS/abcdef0123/slurmdbd_24.11.log
    AWSLogs/PCS/abcdef0123/slurmrestd_25.05.log
    ```

- S3 bucket
  - Ein S3-Bucket-Ausgabepfad folgt dieser Namenskonvention:

    ```
    AWSLogs/${account-id}/PCS/${region}/${cluster_id}/${log_name}/
    ${scheduler_major_version}/yyyy/MM/dd/HH/
    ```

    Example

    ```
    AWSLogs/111111111111/PCS/us-east-2/abcdef0123/slurmctld/25.11/2024/09/01/00/
    AWSLogs/111111111111/PCS/us-east-2/abcdef0123/slurmdbd/24.11/2024/09/01/00/
    AWSLogs/111111111111/PCS/us-east-2/abcdef0123/slurmrestd/25.05/2024/09/01/00/
    ```

  - Ein S3-Objektname folgt dieser Konvention:

    ```
    PCS_${log_name}_${scheduler_major_version}_#{expr date 'event_timestamp', format:
     "yyyy-MM-dd-HH"}_${cluster_id}_${hash}.log
    ```

    Example

    ```
    PCS_slurmctld_25.11_2024-09-01-00_abcdef0123_0123abcdef.log
    ```

Beispiel für Scheduler-Protokolldatensätze

AWS Die PCS-Scheduler-Protokolle sind strukturiert. Sie enthalten Felder wie die Cluster-ID, den Scheduler-Typ, Haupt- und Patch-Versionen sowie die Protokollnachricht, die vom Slurm-Daemon-Prozess ausgegeben wird. Die `node_type` Felder `log_name` und identifizieren, welcher Daemon das Protokoll erstellt hat.

Das folgende Beispiel zeigt einen `slurmctld` Protokolleintrag.

```
{
    "resource_id": "s3431v9rx2",
    "resource_type": "PCS_CLUSTER",
    "event_timestamp": 1721230979,
    "log_level": "info",
    "log_name": "slurmctld",
    "scheduler_type": "slurm",
    "scheduler_major_version": "25.11",
    "scheduler_patch_version": "2",
    "node_type": "controller_primary",
    "message": "[2024-07-17T15:42:58.614+00:00] Running as primary controller\n"
}
```

Das folgende Beispiel zeigt einen `slurmdbd` Protokolldatensatz (Slurm 24.11 und höher).

```
{
    "resource_id": "pcs_bu93qsds2j",
    "resource_type": "PCS_CLUSTER",
    "event_timestamp": 1774485082772,
    "log_level": "info",
    "log_name": "slurmdbd",
    "scheduler_type": "slurm",
    "scheduler_major_version": "25.11",
    "scheduler_patch_version": "2",
    "node_type": "slurmdbd_primary",
    "message": "[2026-03-26T00:31:22.772+00:00] mysql_common: storage token refreshed"
}
```

Das folgende Beispiel zeigt einen `slurmrestd` Protokolldatensatz (Slurm 25.05 und höher).

```
{
    "resource_id": "pcs_bu93qsds2j",
```

```
    "resource_type": "PCS_CLUSTER",
    "event_timestamp": 1774485082772,
    "log_level": "info",
    "log_name": "slurmrestd",
    "scheduler_type": "slurm",
    "scheduler_major_version": "25.05",
    "scheduler_patch_version": "3",
    "node_type": "slurmrestd_primary",
    "message": "[2026-03-26T00:31:22.772+00:00] slurmrestd: Listening on port 6820\n"
}
```


Scheduler-Auditprotokolle in AWS PCS

In den Auditprotokollen des Schedulers werden RPC-Operationen (Remote Procedure Call) aufgezeichnet, die vom Slurm-Controller () und dem Datenbank-Daemon (`slurmctld`) Ihres Clusters verarbeitet werden. `slurmdbd` Das `AUDIT_RPCS:` Präfix in der Protokollnachricht identifiziert diese Protokolle. Sie unterstützen Anwendungsfälle zur Sicherheitsüberprüfung und zur Einhaltung von Vorschriften.

Für Cluster, auf denen Slurm 25.11 und höher ausgeführt wird, stellt AWS PCS Audit-Logs getrennt nach `PCS_SCHEDULER_AUDIT_LOGS` Protokolltyp bereit. Durch diese Trennung können Sie die Aufnahme- und Speicherkosten für Audit-Logs unabhängig von Ihren Betriebsprotokollen kontrollieren, da Audit-Logs bis zu 90% des Scheduler-Protokollvolumens ausmachen können.

> **Note**
>
> Bei Clustern, auf denen Slurm-Versionen vor 25.11 ausgeführt werden, bleiben die Audit-Logs erhalten `PCS_SCHEDULER_LOGS` und der `PCS_SCHEDULER_AUDIT_LOGS` Protokolltyp ist nicht verfügbar. Weitere Informationen zu Scheduler-Logs finden Sie unter. Scheduler loggt sich in AWS PCS ein

Inhalt

Voraussetzungen

Bevor Sie Scheduler-Audit-Logs einrichten können, müssen Sie die folgenden Anforderungen erfüllen:

- Auf Ihrem Cluster muss Slurm 25.11 oder höher ausgeführt werden.
- Der IAM-Prinzipal, der den AWS PCS-Cluster verwaltet, muss die Aktion zulassen. `pcs:AllowVendedLogDeliveryForResource`

Die folgende Beispiel-IAM-Richtlinie gewährt die erforderlichen Berechtigungen.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Sid": "PcsAllowVendedLogsDelivery",
         "Effect": "Allow",
         "Action": ["pcs:AllowVendedLogDeliveryForResource"],
         "Resource": [
            "arn:aws:pcs:*::cluster/*"
         ]
      }
   ]
}
```

Richten Sie Scheduler-Auditprotokolle ein

Sie können Scheduler-Auditprotokolle für Ihren AWS PCS-Cluster mit dem AWS-Managementkonsole oder einrichten. AWS CLI Scheduler-Prüfprotokolle sind optional. AWS PCS liefert sie erst, wenn Sie sie abonnieren.

AWS-Managementkonsole

Um Scheduler-Audit-Logs mit der Konsole einzurichten

1. Öffnen Sie die AWS PCS-Konsole.

2. Klicken Sie im Navigationsbereich auf Cluster.
3. Wählen Sie den Cluster aus, dem Sie Scheduler-Audit-Logs hinzufügen möchten.
4. Wählen Sie auf der Seite mit den Cluster-Details die Registerkarte Logs aus.
5. Wählen Sie unter Scheduler Audit Logs die Option Hinzufügen aus, um bis zu 3 Log-Lieferziele aus CloudWatch Logs, Amazon S3 und Firehose hinzuzufügen.
6. Wählen Sie Protokollzustellungen aktualisieren aus.

AWS CLI

Um Scheduler-Audit-Logs einzurichten mit AWS CLI

1. Erstellen Sie ein Ziel für die Protokollzustellung:

```
aws logs put-delivery-destination --region region \
  --name pcs-audit-logs-destination \
  --delivery-destination-configuration \
  destinationResourceArn=resource-arn
```

Ersetzen Sie:

- *region*— Der AWS-Region Ort, an dem Sie das Ziel erstellen möchten, z. B. `us-east-1`
- *pcs-audit-logs-destination*— Ein Name für das Ziel
- *resource-arn*— Der Amazon-Ressourcenname (ARN) einer CloudWatch Logs-Protokollgruppe, eines S3-Buckets oder eines Firehose-Lieferstreams.

Weitere Informationen finden Sie PutDeliveryDestinationin der Amazon CloudWatch Logs API-Referenz.

2. Legen Sie den PCS-Cluster als Quelle für die Protokollzustellung fest:

```
aws logs put-delivery-source --region region \
  --name cluster-audit-logs-source-name \
  --resource-arn cluster-arn \
  --log-type PCS_SCHEDULER_AUDIT_LOGS
```

Ersetzen Sie:

- *region*— Die AWS-Region Ihres Clusters, wie `us-east-1`

- *cluster-audit-logs-source-name*— Ein Name für die Quelle
- *cluster-arn*— der ARN Ihres AWS PCS-Clusters

Weitere Informationen finden Sie PutDeliverySourcein der Amazon CloudWatch Logs API-Referenz.

3. Connect die Lieferquelle mit dem Lieferziel:

```
aws logs create-delivery --region region \
  --delivery-source-name cluster-audit-logs-source \
  --delivery-destination-arn destination-arn
```

Ersetzen Sie:

- *region*— Die AWS-Region, wie `us-east-1`
- *cluster-audit-logs-source*— Der Name Ihrer Lieferquelle
- *destination-arn*— Die ARN Ihres Lieferziels

Weitere Informationen finden Sie CreateDeliveryin der Amazon CloudWatch Logs API-Referenz.

Die Pfade und Namen von Log-Streams werden von Scheduler geprüft

Der Pfad und der Name der AWS PCS-Scheduler-Audit-Logs hängen vom Zieltyp ab.

- CloudWatch Protokolle
  - Ein CloudWatch Logs-Stream folgt dieser Namenskonvention.

```
AWSLogs/PCS/${cluster_id}/${log_name}_${scheduler_major_version}_audit.log
```

Wo `${log_name}` ist `slurmctld` oder`slurmdbd`.

Example

```
AWSLogs/PCS/abcdef0123/slurmctld_25.11_audit.log
AWSLogs/PCS/abcdef0123/slurmdbd_25.11_audit.log
```

- S3 bucket
  - Ein S3-Bucket-Ausgabepfad folgt dieser Namenskonvention:

    ```
    AWSLogs/${account-id}/PCS/${region}/${cluster_id}/scheduler_audit/${log_name}/yyyy/
    MM/dd/HH/
    ```

    Example

    ```
    AWSLogs/111111111111/PCS/us-east-2/abcdef0123/scheduler_audit/
    slurmctld/2026/03/01/00/
    AWSLogs/111111111111/PCS/us-east-2/abcdef0123/scheduler_audit/
    slurmdbd/2026/03/01/00/
    ```

Beispiel für einen Scheduler-Audit-Logeintrag

AWS Die Auditprotokolle von PCS Scheduler sind strukturiert. Sie verwenden dasselbe Schema wie Scheduler-Logs, wobei die Log-Meldung das Präfix enthält. `AUDIT_RPCS:` Hier ist ein Beispiel von`slurmctld`.

```
{
    "resource_id": "pcs_bu93qsds2j",
    "resource_type": "PCS_CLUSTER",
    "event_timestamp": 1774481175953,
    "log_level": "info",
    "log_name": "slurmctld",
    "scheduler_type": "slurm",
    "scheduler_major_version": "25.11",
    "scheduler_patch_version": "2",
    "node_type": "controller_primary",
    "message": "[2026-01-21T08:19:26.692+00:00] AUDIT_RPCS: [slurmctld-
primary:6817(fd:18)] msg_type=REQUEST_PARTITION_INFO uid=0 client=[10.0.76.95:56918]\n"
}
```

Hier ist ein Beispiel von`slurmdbd`.

```
{
    "resource_id": "pcs_bu93qsds2j",
    "resource_type": "PCS_CLUSTER",
    "event_timestamp": 1774485082772,
    "log_level": "info",
```

```
    "log_name": "slurmdbd",
    "scheduler_type": "slurm",
    "scheduler_major_version": "25.11",
    "scheduler_patch_version": "2",
    "node_type": "slurmdbd_primary",
    "message": "[2026-01-21T08:19:26.692+00:00] AUDIT_RPCS: msg_type=DBD_GET_CLUSTERS
 uid=0 client=[28.5.0.18:36658] protocol=11008\n"
}
```


Verhalten des Audit-Logs nach Slurm-Version

In der folgenden Tabelle wird beschrieben, wie Audit-Logs je nach der Slurm-Version, die auf Ihrem Cluster ausgeführt wird, bereitgestellt werden.

Slurm-Version	**PCS_SCHEDULER_LOGS** enthält	**PCS_SCHEDULER_AUDIT_LOGS** verfügbar
Vor 25.11	Alle Protokolle, einschließlich Audit-Logs	Nein
25.11 und später	Nur Betriebsprotokolle (Auditprotokolle wurden entfernt)	Ja (Opt-In)

Überwachung des AWS Parallel Computing Service mit Amazon CloudWatch

Amazon überwacht CloudWatch den Zustand und die Leistung Ihres AWS Parallel Computing Service (AWS PCS) -Clusters, indem es in regelmäßigen Abständen Metriken aus dem Cluster sammelt. Diese Metriken werden gespeichert, sodass Sie auf historische Daten zugreifen und Einblicke in die Leistung Ihres Clusters im Laufe der Zeit gewinnen können.

CloudWatch ermöglicht es Ihnen auch, die von AWS PCS gestarteten EC2-Instances zu überwachen, um Ihre Skalierungsanforderungen zu erfüllen. Sie können zwar die Protokolle laufender Instances überprüfen, CloudWatch Metriken und Protokolldaten werden jedoch in der Regel gelöscht, sobald Instances beendet werden. Sie können den CloudWatch Agenten auf Instances jedoch mithilfe einer EC2-Startvorlage so konfigurieren, dass Metriken und Protokolle auch nach dem Beenden der Instance beibehalten werden, was eine langfristige Überwachung und Analyse ermöglicht.

Erkunden Sie die Themen in diesem Abschnitt, um mehr über die Überwachung von AWS PCS zu erfahren. CloudWatch

Themen

- Überwachung von AWS PCS-Metriken mit CloudWatch
- Überwachung von AWS PCS-Instanzen mit Amazon CloudWatch

Überwachung von AWS PCS-Metriken mit CloudWatch

Sie können den Zustand des AWS PCS-Clusters mithilfe von Amazon CloudWatch überwachen. Amazon sammelt Daten aus Ihrem Cluster und wandelt sie in Metriken nahezu in Echtzeit um. Diese Statistiken werden für einen Zeitraum von 15 Monaten aufbewahrt, sodass Sie auf historische Informationen zugreifen und sich einen besseren Überblick über die Leistung Ihres Clusters verschaffen können. Cluster-Metriken werden CloudWatch in Abständen von 1 Minute an gesendet. Weitere Informationen zu CloudWatch finden Sie unter Was ist Amazon CloudWatch? im CloudWatch Amazon-Benutzerhandbuch.

AWS PCS veröffentlicht die folgenden Metriken im AWS/PCSNamespace in CloudWatch. Sie haben eine einzige Dimension,`ClusterId`.

Name	Description	Einheiten
ActualCapacity	IdleCapacity + UtilizedCapacity	Anzahl
CapacityUtilization	UtilizedCapacity / ActualCapacity	Anzahl
DesiredCapacity	ActualCapacity + PendingCapacity	Anzahl
IdleCapacity	Anzahl der Instances, die ausgeführt werden, aber keinen Jobs zugewiesen sind	Anzahl
UtilizedCapacity	Anzahl der Instances, die ausgeführt werden und Jobs zugewiesen sind	Anzahl

Überwachung von AWS PCS-Instanzen mit Amazon CloudWatch

AWS PCS startet Amazon EC2 EC2-Instances nach Bedarf, um die in Ihren PCS-Rechenknotengruppen definierten Skalierungsanforderungen zu erfüllen. Sie können diese Instances mit Amazon überwachen, während sie ausgeführt werden CloudWatch. Sie können die Protokolle laufender Instances einsehen, indem Sie sich bei ihnen anmelden und interaktive Befehlszeilentools verwenden. Standardmäßig werden CloudWatch Metrikdaten jedoch nur für einen begrenzten Zeitraum aufbewahrt, sobald eine Instance beendet wurde. Instance-Protokolle werden normalerweise zusammen mit den EBS-Volumes gelöscht, die die Instance unterstützen. Um Metriken oder Protokolldaten der von PCS gestarteten Instances nach deren Beendigung beizubehalten, können Sie den CloudWatch Agenten auf Ihren Instances mit einer EC2-Startvorlage konfigurieren. Dieses Thema bietet einen Überblick über die Überwachung laufender Instances und enthält Beispiele für die Konfiguration persistenter Instance-Metriken und Logs.

Überwachung laufender Instanzen

Suchen nach AWS-PCS-Instanzen

Um von PCS gestartete Instances zu überwachen, suchen Sie nach den laufenden Instances, die einem Cluster oder einer Rechenknotengruppe zugeordnet sind. Überprüfen Sie dann in der EC2-Konsole für eine bestimmte Instance die Abschnitte Status und Alarme sowie Überwachung. Wenn der Anmeldezugriff für diese Instances konfiguriert ist, können Sie eine Verbindung zu ihnen herstellen und verschiedene Protokolldateien auf den Instances überprüfen. Weitere Informationen zur Identifizierung der Instanzen, die von PCS verwaltet werden, finden Sie unterSuchen nach Rechenknotengruppeninstanzen in AWS PCS.

Aktivierung detaillierter Metriken

Standardmäßig werden Instanzmetriken in Intervallen von 5 Minuten erfasst. Um Metriken in Intervallen von einer Minute zu erfassen, aktivieren Sie die detaillierte CloudWatch Überwachung in Ihrer Vorlage für den Start von Compute-Knotengruppen. Weitere Informationen finden Sie unter Aktivieren Sie die detaillierte CloudWatch Überwachung.

Konfiguration persistenter Instanzmetriken und -protokolle

Sie können die Metriken und Protokolle Ihrer Instances behalten, indem Sie den CloudWatch Amazon-Agenten auf ihnen installieren und konfigurieren. Dies besteht aus drei Hauptschritten:

1. Erstellen Sie eine CloudWatch Agentenkonfiguration.
2. Speichern Sie die Konfiguration dort, wo sie von PCS-Instanzen abgerufen werden kann.

3. Schreiben Sie eine EC2-Startvorlage, die die CloudWatch Agentsoftware installiert, Ihre Konfiguration abruft und den CloudWatch Agenten anhand der Konfiguration startet.

Weitere Informationen finden Sie unter Erfassung von Metriken, Protokollen und Traces mit dem CloudWatch Agenten im CloudWatch Amazon-Benutzerhandbuch und Verwenden von Amazon EC2 EC2-Startvorlagen mit AWS PCS.

Erstellen Sie eine CloudWatch Agentenkonfiguration

Bevor Sie den CloudWatch Agenten auf Ihren Instances bereitstellen, müssen Sie eine JSON-Konfigurationsdatei generieren, die die zu erfassenden Metriken, Logs und Traces spezifiziert. Konfigurationsdateien können mit einem Assistenten oder manuell mithilfe eines Texteditors erstellt werden. Die Konfigurationsdatei wird für diese Demonstration manuell erstellt.

Erstellen Sie auf einem Computer, auf dem die AWS-CLI installiert ist, eine CloudWatch Konfigurationsdatei namens config.json mit dem folgenden Inhalt. Sie können auch die folgende URL verwenden, um eine Kopie der Datei herunterzuladen.

```
https://aws-hpc-recipes.s3.amazonaws.com/main/recipes/pcs/cloudwatch/assets/config.json
```

Hinweise

- Die Protokollpfade in der Beispieldatei beziehen sich auf Amazon Linux 2023. Wenn Ihre Instances ein anderes Basisbetriebssystem verwenden, ändern Sie die Pfade entsprechend.
- Um andere Protokolle zu erfassen, fügen Sie weitere Einträge unter hinzu`collect_list`.
- Bei den Werten in `{brackets}` handelt es sich um Vorlagenvariablen. Die vollständige Liste der unterstützten Variablen finden Sie unter Manuelles Erstellen oder Bearbeiten der CloudWatch Agentenkonfigurationsdatei im CloudWatch Amazon-Benutzerhandbuch.
- Sie können wählen, `metrics` ob Sie diese Informationstypen weglassen `logs` oder nicht sammeln möchten.

```
{
    "agent": {
        "metrics_collection_interval": 60
    },
    "logs": {
        "logs_collected": {
            "files": {
```

```
                "collect_list": [
                    {
                        "file_path": "/var/log/cloud-init.log",
                        "log_group_class": "STANDARD",
                        "log_group_name": "/PCSLogs/instances",
                        "log_stream_name": "{instance_id}.cloud-init.log",
                        "retention_in_days": 30
                    },
                    {
                        "file_path": "/var/log/cloud-init-output.log",
                        "log_group_class": "STANDARD",
                        "log_stream_name": "{instance_id}.cloud-init-output.log",
                        "log_group_name": "/PCSLogs/instances",
                        "retention_in_days": 30
                    },
                    {
                        "file_path": "/var/log/amazon/pcs/bootstrap.log",
                        "log_group_class": "STANDARD",
                        "log_stream_name": "{instance_id}.bootstrap.log",
                        "log_group_name": "/PCSLogs/instances",
                        "retention_in_days": 30
                    },
                    {
                        "file_path": "/var/log/slurmd.log",
                        "log_group_class": "STANDARD",
                        "log_stream_name": "{instance_id}.slurmd.log",
                        "log_group_name": "/PCSLogs/instances",
                        "retention_in_days": 30
                    },
                    {
                        "file_path": "/var/log/messages",
                        "log_group_class": "STANDARD",
                        "log_stream_name": "{instance_id}.messages",
                        "log_group_name": "/PCSLogs/instances",
                        "retention_in_days": 30
                    },
                    {
                        "file_path": "/var/log/secure",
                        "log_group_class": "STANDARD",
                        "log_stream_name": "{instance_id}.secure",
                        "log_group_name": "/PCSLogs/instances",
                        "retention_in_days": 30
                    }
                ]
```

```
            }
        }
    },
    "metrics": {
        "aggregation_dimensions": [
            [
                "InstanceId"
            ]
        ],
        "append_dimensions": {
            "AutoScalingGroupName": "${aws:AutoScalingGroupName}",
            "ImageId": "${aws:ImageId}",
            "InstanceId": "${aws:InstanceId}",
            "InstanceType": "${aws:InstanceType}"
        },
        "metrics_collected": {
            "cpu": {
                "measurement": [
                    "cpu_usage_idle",
                    "cpu_usage_iowait",
                    "cpu_usage_user",
                    "cpu_usage_system"
                ],
                "metrics_collection_interval": 60,
                "resources": [
                    "*"
                ],
                "totalcpu": false
            },
            "disk": {
                "measurement": [
                    "used_percent",
                    "inodes_free"
                ],
                "metrics_collection_interval": 60,
                "resources": [
                    "*"
                ]
            },
            "diskio": {
                "measurement": [
                    "io_time"
                ],
                "metrics_collection_interval": 60,
```

```
                "resources": [
                    "*"
                ]
            },
            "mem": {
                "measurement": [
                    "mem_used_percent"
                ],
                "metrics_collection_interval": 60
            },
            "swap": {
                "measurement": [
                    "swap_used_percent"
                ],
                "metrics_collection_interval": 60
            }
        }
    }
}
```

Diese Datei weist den CloudWatch Agenten an, mehrere Dateien zu überwachen, was bei der Diagnose von Fehlern bei Instance-Bootstrapping, Authentifizierung und Anmeldung sowie bei anderen Problembehandlungsdomänen hilfreich sein kann. Dazu zählen:

- `/var/log/cloud-init.log`— Ausgabe aus der Anfangsphase der Instanzkonfiguration
- `/var/log/cloud-init-output.log`— Ausgabe von Befehlen, die während der Instanzkonfiguration ausgeführt werden
- `/var/log/amazon/pcs/bootstrap.log`— Ausgabe von PCS-specific Vorgängen, die während der Instanzkonfiguration ausgeführt werden
- `/var/log/slurmd.log`— Ausgabe vom Daemon slurmd des Slurm-Workload-Managers
- `/var/log/messages`— Systemnachrichten vom Kernel, von Systemdiensten und Anwendungen
- `/var/log/secure`— Protokolle im Zusammenhang mit Authentifizierungsversuchen wie SSH, Sudo und anderen Sicherheitsereignissen

Die Protokolldateien werden an eine CloudWatch Protokollgruppe mit dem Namen gesendet. `/PCSLogs/instances` Die Protokollstreams sind eine Kombination aus der Instanz-ID und dem Basisnamen der Protokolldatei. Die Protokollgruppe hat eine Aufbewahrungszeit von 30 Tagen.

Darüber hinaus weist die Datei den CloudWatch Agenten an, mehrere allgemeine Messwerte zu sammeln und sie nach Instanz-ID zu aggregieren.

Speichern Sie die Konfiguration

Die CloudWatch Agenten-Konfigurationsdatei muss an einem Ort gespeichert werden, auf den PCS-Compute-Knoteninstanzen zugegriffen werden kann. Dafür gibt es zwei gängige Methoden. Sie können es in einen Amazon S3 S3-Bucket hochladen, auf den Ihre Compute-Knotengruppen-Instances über ihr Instance-Profil Zugriff haben. Alternativ können Sie es als SSM-Parameter im Amazon Systems Manager Parameter Store speichern.

Laden Sie es in einen S3-Bucket hoch

Verwenden Sie die folgenden AWS-CLI-Befehle, um Ihre Datei in S3 zu speichern. Nehmen Sie vor der Ausführung des Befehls die folgenden Ersetzungen vor:

- Ersetzen Sie es *amzn-s3-demo-bucket* durch Ihren eigenen S3-Bucket-Namen

Erstellen Sie zunächst (dies ist optional, wenn Sie über einen vorhandenen Bucket verfügen) einen Bucket, der Ihre Konfigurationsdatei (en) enthält.

```
aws s3 mb s3://amzn-s3-demo-bucket
```

Laden Sie als Nächstes die Datei in den Bucket hoch.

```
aws s3 cp ./config.json s3://amzn-s3-demo-bucket/
```

Als SSM-Parameter speichern

Verwenden Sie den folgenden Befehl, um Ihre Datei als SSM-Parameter zu speichern. Nehmen Sie vor der Ausführung des Befehls die folgenden Ersetzungen vor:

- *region-code*Ersetzen Sie durch die AWS-Region, in der Sie mit AWS PCS arbeiten.
- (Optional) *AmazonCloudWatch-PCS* Ersetzen Sie den Parameter durch Ihren eigenen Namen. Beachten Sie, dass Sie, wenn Sie das Präfix des Namens von `AmazonCloudWatch-` ändern, ausdrücklich Lesezugriff auf den SSM-Parameter in Ihrem Knotengruppen-Instanzprofil hinzufügen müssen.

```
aws ssm put-parameter \
```

```
    --region region-code \
    --name "AmazonCloudWatch-PCS" \
    --type String \
    --value file://config.json
```

Schreiben Sie eine EC2-Startvorlage

Die spezifischen Details für die Startvorlage hängen davon ab, ob Ihre Konfigurationsdatei in S3 oder SSM gespeichert ist.

Verwenden Sie eine in S3 gespeicherte Konfiguration

Dieses Skript installiert den CloudWatch Agenten, importiert eine Konfigurationsdatei aus einem S3-Bucket und startet den CloudWatch Agenten damit. Ersetzen Sie die folgenden Werte in diesem Skript durch Ihre eigenen Details:

- *amzn-s3-demo-bucket*— Der Name eines S3-Buckets, aus dem Ihr Konto lesen kann
- */config.json*— Pfad relativ zum S3-Bucket-Root, in dem die Konfiguration gespeichert ist

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="==MYBOUNDARY=="

--==MYBOUNDARY==
Content-Type: text/cloud-config; charset="us-ascii"

packages:
- amazon-cloudwatch-agent

runcmd:
- aws s3 cp s3://amzn-s3-demo-bucket/config.json /etc/s3-cw-config.json
- /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m
 ec2 -s -c file://etc/s3-cw-config.json

--==MYBOUNDARY==--
```

Das IAM-Instanzprofil für die Knotengruppe muss Zugriff auf den Bucket haben. Hier ist ein Beispiel für eine IAM-Richtlinie für den Bucket im obigen Benutzerdatenskript.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Beachten Sie außerdem, dass die Instances ausgehenden Datenverkehr zum S3 und CloudWatch zu den Endpunkten zulassen müssen. Dies kann je nach Ihrer Clusterarchitektur mithilfe von Sicherheitsgruppen oder VPC-Endpunkten erreicht werden.

Verwenden Sie eine in SSM gespeicherte Konfiguration

Dieses Skript installiert den CloudWatch Agenten, importiert eine Konfigurationsdatei aus einem SSM-Parameter und startet den CloudWatch Agenten damit. Ersetzen Sie die folgenden Werte in diesem Skript durch Ihre eigenen Details:

- (Optional) *AmazonCloudWatch-PCS* Ersetzen Sie den Parameter durch Ihren eigenen Namen.

```
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="==MYBOUNDARY=="

--==MYBOUNDARY==
Content-Type: text/cloud-config; charset="us-ascii"

packages:
- amazon-cloudwatch-agent
```

```
runcmd:
- /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m
 ec2 -s -c ssm:AmazonCloudWatch-PCS

--==MYBOUNDARY==--
```

Der IAM-Instanzrichtlinie für die Knotengruppe muss das CloudWatchAgentServerPolicyangehängt sein.

Wenn Ihr Parametername nicht mit beginnt, müssen `AmazonCloudWatch-` Sie ausdrücklich Lesezugriff auf den SSM-Parameter in Ihrem Knotengruppen-Instanzprofil hinzufügen. Hier ist ein Beispiel für eine IAM-Richtlinie, die dies für das Präfix veranschaulicht. *DOC-EXAMPLE-PREFIX*

JSON

```
{
  "Version":"2012-10-17",
  "Statement" : [
    {
      "Sid" : "CustomCwSsmMParamReadOnly",
      "Effect" : "Allow",
      "Action" : [
        "ssm:GetParameter"
      ],
      "Resource" : "arn:aws:ssm:*:*:parameter/DOC-EXAMPLE-PREFIX*"
    }
  ]
}
```

Beachten Sie außerdem, dass die Instances ausgehenden Datenverkehr zum SSM und zu den Endpunkten zulassen müssen. CloudWatch Dies kann je nach Ihrer Clusterarchitektur mithilfe von Sicherheitsgruppen oder VPC-Endpunkten erreicht werden.

Protokollieren von API-Aufrufen für den AWS Parallel Computing Service mit AWS CloudTrail

AWS PCS ist in einen Dienst integriert AWS CloudTrail, der eine Aufzeichnung der Aktionen bereitstellt, die von einem Benutzer, einer Rolle oder einem AWS Dienst in AWS PCS ausgeführt wurden. CloudTrail erfasst alle API-Aufrufe für AWS PCS als Ereignisse. Zu den erfassten Aufrufen

gehören Aufrufe von der AWS PCS-Konsole und Codeaufrufen für die AWS PCS-API-Operationen. Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Übermittlung von CloudTrail Ereignissen an einen Amazon S3 S3-Bucket aktivieren, einschließlich Ereignissen für AWS PCS. Wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse trotzdem in der CloudTrail Konsole im Ereignisverlauf anzeigen. Anhand der von gesammelten Informationen können Sie die Anfrage CloudTrail, die an AWS PCS gestellt wurde, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere Details ermitteln.

Weitere Informationen CloudTrail dazu finden Sie im AWS CloudTrail Benutzerhandbuch.

AWS PCS-Informationen in CloudTrail

CloudTrail ist auf Ihrem aktiviert AWS-Konto , wenn Sie das Konto erstellen. Wenn in AWS PCS eine Aktivität auftritt, wird diese Aktivität zusammen mit anderen CloudTrail AWS Serviceereignissen im Ereignisverlauf in einem Ereignis aufgezeichnet. Sie können aktuelle Ereignisse in Ihrem anzeigen, suchen und herunterladen AWS-Konto. Weitere Informationen finden Sie unter Ereignisse mit dem CloudTrail Ereignisverlauf anzeigen.

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem System AWS-Konto, einschließlich Ereignisse für AWS PCS, erstellen Sie einen Trail. Ein Trail ermöglicht CloudTrail die Übermittlung von Protokolldateien an einen Amazon S3 S3-Bucket. Wenn Sie einen Trail in der Konsole anlegen, gilt dieser für alle AWS-Regionen-Regionen. Der Trail protokolliert Ereignisse aus allen Regionen der AWS Partition und übermittelt die Protokolldateien an den von Ihnen angegebenen Amazon S3 S3-Bucket. Darüber hinaus können Sie andere AWS Dienste konfigurieren, um die in den CloudTrail Protokollen gesammelten Ereignisdaten weiter zu analysieren und darauf zu reagieren. Weitere Informationen finden Sie hier:

- Übersicht zum Erstellen eines Trails
- CloudTrail unterstützte Dienste und Integrationen
- Konfiguration von Amazon SNS SNS-Benachrichtigungen für CloudTrail
- Empfangen von CloudTrail Protokolldateien aus mehreren Regionen und Empfangen von CloudTrail Protokolldateien von mehreren Konten

Alle AWS PCS-Aktionen werden von der AWS Parallel Computing Service API-Referenz protokolliert CloudTrail und sind in dieser dokumentiert. Beispielsweise generieren Aufrufe der `DeleteCluster` Aktionen `CreateComputeNodeGroupUpdateQueue`, und Einträge in den CloudTrail Protokolldateien.

Jeder Ereignis- oder Protokolleintrag enthält Informationen zu dem Benutzer, der die Anforderung generiert hat. Die Identitätsinformationen unterstützen Sie bei der Ermittlung der folgenden Punkte:

- Ob die Anfrage mit Root- oder AWS Identity and Access Management (IAM-) Benutzeranmeldedaten gestellt wurde.
- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anfrage von einem anderen AWS Dienst gestellt wurde.

Weitere Informationen finden Sie unter CloudTrail -Element userIdentity.

Grundlegendes zu CloudTrail Protokolldateieinträgen von AWS PCS

Ein Trail ist eine Konfiguration, die die Übertragung von Ereignissen als Protokolldateien an einen von Ihnen angegebenen S3-Bucket ermöglicht. CloudTrail Protokolldateien enthalten einen oder mehrere Protokolleinträge. Ein Ereignis stellt eine einzelne Anforderung aus einer beliebigen Quelle dar und enthält Informationen über die angeforderte Aktion, Datum und Uhrzeit der Aktion, Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordneter Stack-Trace der öffentlichen API-Aufrufe, sodass sie nicht in einer bestimmten Reihenfolge angezeigt werden.

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag für eine `CreateQueue` Aktion.


```
{
    "eventVersion": "1.09",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE:admin",
        "arn": "arn:aws:sts::012345678910:assumed-role/Admin/admin",
        "accountId": "012345678910",
        "accessKeyId": "ASIAY36PTPIEXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAY36PTPIEEXAMPLE",
                "arn": "arn:aws:iam::012345678910:role/Admin",
                "accountId": "012345678910",
                "userName": "Admin"
            },
            "attributes": {
                "creationDate": "2024-07-16T17:05:51Z",
```

```
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2024-07-16T17:13:09Z",
    "eventSource": "pcs.amazonaws.com",
    "eventName": "CreateQueue",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36",
    "requestParameters": {
        "clientToken": "c13b7baf-2894-42e8-acec-example",
        "clusterIdentifier": "abcdef0123",
        "computeNodeGroupConfigurations": [
            {
                "computeNodeGroupId": "abcdef0123"
            }
        ],
        "queueName": "all"
    },
    "responseElements": {
        "queue": {
            "arn": "arn:aws:pcs:us-east-1:609783872011:cluster/abcdef0123/queue/
abcdef0123",
            "clusterId": "abcdef0123",
            "computeNodeGroupConfigurations": [
                {
                    "computeNodeGroupId": "abcdef0123"
                }
            ],
            "createdAt": "2024-07-16T17:13:09.276069393Z",
            "id": "abcdef0123",
            "modifiedAt": "2024-07-16T17:13:09.276069393Z",
            "name": "all",
            "status": "CREATING"
        }
    },
    "requestID": "a9df46d7-3f6d-43a0-9e3f-example",
    "eventID": "7ab18f88-0040-47f5-8388-example",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "012345678910",
```

```
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.3",
        "cipherSuite": "TLS_AES_128_GCM_SHA256",
        "clientProvidedHostHeader": "pcs.us-east-1.amazonaws.com"
    },
    "sessionCredentialFromConsole": "true"
}
```

Endpunkte und Servicekontingenten für AWS STK.

In den folgenden Abschnitten werden die Endpunkte und Dienstkontingente für AWS Parallel Computing Service (AWS PCS) beschrieben. Servicekontingenten, früher als Limits bezeichnet, sind die maximale Anzahl von Serviceressourcen oder Vorgängen für Ihre AWS-Konto.

Ihr AWS-Konto hat Standardkontingente für jeden AWS Dienst. Sofern nicht anders angegeben, gilt für jedes Kontingent Region-specific. Sie können Erhöhungen für einige Kontingente beantragen und andere Kontingente können nicht erhöht werden.

Weitere Informationen finden Sie unter AWS Service Quotas in der Allgemeinen AWS -Referenz.

Inhalt

Service-Endpunkte

Name der Region	Region	Endpunkt	Protocol (Protokoll)
USA Ost (Ohio)	us-east-2	pcs.us-east-2.amazonaws.com pcs-fips.us-east-2.amazonaws.com pcs-fips.us-east-2.api.aws pcs.us-east-2.api.aws	HTTPS
USA Ost (Nord-Virginia)	us-east-1	pcs.us-east-1.amazonaws.com	HTTPS

Name der Region	Region	Endpunkt	Protocol (Protokoll)
		pcs-fips.us-east-1.amazonaws.com pcs-fips.us-east-1.api.aws pcs.us-east-1.api.aws	
USA West (Oregon)	us-west-2	pcs.us-west-2.amazonaws.com pcs-fips.us-west-2.amazonaws.com pcs-fips.us-west-2.api.aws pcs.us-west-2.api.aws	HTTPS
Asien-Pazifik (Mumbai)	ap-south-1	pcs.ap-south-1.amazonaws.com pcs.ap-south-1.api.aws	HTTPS
Asien-Pazifik (Singapur)	ap-southeast-1	pcs.ap-southeast-1.amazonaws.com pcs.ap-southeast-1.api.aws	HTTPS
Asien-Pazifik (Sydney)	ap-southeast-2	pcs.ap-southeast-2.amazonaws.com pcs.ap-southeast-2.api.aws	HTTPS

Name der Region	Region	Endpunkt	Protocol (Protokoll)
Asien-Pazifik (Tokio)	ap-northeast-1	pcs.ap-northeast-1.amazonaws.com pcs.ap-northeast-1.api.aws	HTTPS
Asia Pacific (Osaka)	ap-northeast-3	pcs.ap-northeast-3.amazonaws.com pcs.ap-northeast-3.api.aws	HTTPS
Europa (Frankfurt)	eu-central-1	pcs.eu-central-1.amazonaws.com pcs.eu-central-1.api.aws	HTTPS
Europa (Irland)	eu-west-1	pcs.eu-west-1.amazonaws.com pcs.eu-west-1.api.aws	HTTPS
Europa (London)	eu-west-2	pcs.eu-west-2.amazonaws.com pcs.eu-west-2.api.aws	HTTPS
Europa (Paris)	eu-west-3	pcs.eu-west-3.amazonaws.com pcs-eu-west-3.api.aws	HTTPS
Europa (Milan)	eu-south-1	pcs.eu-south-1.amazonaws.com pcs-eu-süd-1.api.aws	HTTPS

Name der Region	Region	Endpunkt	Protocol (Protokoll)
Europa (Spain)	eu-south-2	pcs.eu-south-2.amazonaws.com pcs-eu-süd-2.api.aws	HTTPS
Europa (Stockholm)	eu-north-1	pcs.eu-north-1.amazonaws.com pcs.eu-north-1.api.aws	HTTPS
Südamerika (São Paulo)	sa-east-1	pcs.sa-east-1.amazonaws.com pcs.sa-east-1.api.aws	HTTPS
AWS GovCloud (US-East)	us-gov-east-1	pcs.us-gov-east-1.amazonaws.com pcs-fips.us-gov-east-1.amazonaws.com pcs-fips.us-gov-east-1.api.aws pcs.us-gov-east-1.api.aws	HTTPS

Name der Region	Region	Endpunkt	Protocol (Protokoll)
AWS GovCloud (US-West)	us-gov-west-1	pcs.us-gov-west-1.amazonaws.com pcs-fips.us-gov-west-1.amazonaws.com pcs-fips.us-gov-west-1.api.aws pcs.us-gov-west-1.api.aws	HTTPS

Servicekontingente

Name	Standard	Einstellbar	Beschreibung
Cluster	5	Ja	Die maximale Anzahl von Clustern pro. AWS-Region

> **Note**
>
> Die Standardwerte sind die anfänglichen Kontingente, die von festgelegt wurden AWS. Diese Standardwerte sind unabhängig von den tatsächlich angewendeten Kontingentwerten und den maximal möglichen Servicekontingenten. Weitere Informationen finden Sie unter Terminologie in Service Quotas im Service Quotas User Guide.

Diese Dienstkontingente sind unter AWS Parallel Computing Service (PCS) in der aufgeführt AWS-Managementkonsole. Informationen zum Beantragen einer Kontingenterhöhung für Werte, die als anpassbar angezeigt werden, finden Sie unter Eine Kontingenterhöhung beantragen im Benutzerhandbuch für Servicekontingente.

> **Important**
>
> Denken Sie daran, die aktuelle AWS-Region Einstellung in der zu überprüfen AWS-Managementkonsole.

Interne Kontingente

Die folgenden Kontingente sind intern und nicht anpassbar.

Name	Standard	Einstellbar	Beschreibung
Gleichzeitige Clustererstellung	1	Nein	Die maximale Anzahl von Clustern im `Creating` Bundesstaat pro. AWS-Region
Knotengruppen pro Cluster berechnen	10	Nein	Die maximale Anzahl von Rechenknotengruppen pro Cluster.
Warteschlangen pro Cluster	10	Nein	Die maximale Anzahl von Warteschlangen pro Cluster.

Relevante Kontingente für andere AWS service

AWS PCS nutzt andere AWS Dienste. Ihre Servicekontingenten für diese Dienste wirken sich auf Ihre Nutzung von AWS PCS aus.

Amazon EC2-Servicekontingente, die sich auswirken AWS STK.

- Spot-Instance-Anfragen
- On-Demand-Instances ausführen
- Startvorlagen
- Startvorlagenversionen

- Amazon EC2 EC2-API-Anfragen

Weitere Informationen finden Sie unter Amazon EC2-Servicekontingente im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Behebung von Problemen in AWS Dienst für parallele Datenverarbeitung

Die folgenden Themen enthalten Anleitungen zur Behebung einiger Probleme, die bei AWS PCS auftreten können.

- Cluster-Updates
- Probleme mit dem Bootstrap von Compute-Knoten
- Benutzerdefinierte Slurm-Einstellungen
- EC2-Instances wurden nach dem Neustart beendet
- Identität und Zugriff
- MaxJobCount Limit Job Stelleneinreichungen
- Probleme beim Neustart von Slurm

Eine EC2-Instanz in AWS PCS wird nach dem Neustart beendet und ersetzt

Überblick über das Problem

Nach dem Neustart einer EC2-Instance in einer Compute-Knotengruppe beendet AWS PCS die Instanz automatisch und ersetzt sie.

Warum passiert das

AWS PCS unterstützt keine Instanzneustarts. Wenn eine EC2-Instance neu gestartet wird, betrachtet AWS PCS die Instance als fehlerhaft und ersetzt sie. Wenn AWS PCS Ihre Instances kontinuierlich beendet und ersetzt, kann das daran liegen, dass Ihre Instances nach dem Start neu gestartet werden. Einige Beispiele hierfür sind automatische Neustarts auf der EC2-Instance (z. B. ein automatischer Neustart nach dem Patchen), Automatisierung außerhalb der EC2-Instance (z. B. eine Netzwerkverwaltungsanwendung), ein anderer AWS Dienst (z. B. AWS Systems Manager) oder ein manueller Neustart durch eine Person.

Vorgehensweise

Sie können in Ihren `slurmctld` `slurmd` OP-Protokollen nachsehen, ob Ihre Instance neu gestartet wurde. Weitere Informationen erhalten Sie unter Scheduler loggt sich in AWS PCS ein

und Überwachung von AWS PCS-Instanzen mit Amazon CloudWatch. Der folgende `slurmctld` Beispielprotokolleintrag gibt an, dass die Instanz neu gestartet wurde:

Example

```
[2024-09-12T06:42:50.393+00:00] validate_node_specs: Node Login-1 unexpectedly rebooted
 boot_time=1726123354 last response=1726123285
```

Neustart aufgrund von Patches

Nach der Installation von Patches ist häufig ein Neustart erforderlich. Wenden Sie Patches nicht direkt auf eine EC2-Instance an, die Teil einer AWS PCS-Rechenknotengruppe ist. Wenn Sie Ihre EC2-Instances patchen müssen, sollten Sie Ihre Patches auf ein aktualisiertes Amazon Machine Image (AMI) anwenden und Ihre Rechenknotengruppen aktualisieren, um das aktualisierte AMI zu verwenden. Neue EC2-Instances, die AWS PCS für diese Rechenknotengruppen startet, verwenden das aktualisierte (gepatchte) AMI. Weitere Informationen finden Sie unter Benutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS.

Beheben Sie Probleme mit dem Bootstrap und der Registrierung von Rechenknoten in AWS STK.

Wenn Compute-Knoten beim Bootstrap oder bei der Registrierung nicht ordnungsgemäß bei Ihrem AWS PCS-Cluster funktionieren, können die folgenden Symptome auftreten:

- Jobs werden nicht gestartet
- Sie können keine Verbindung zu Instanzen herstellen in AWS Systems Manager
- Instanzen werden unerwartet heruntergefahren
- Instanzen werden kontinuierlich ersetzt

Diese Fehler können durch Probleme beim Start der EC2-Instance oder beim Bootstrap-Prozess des AWS PCS-Compute-Knotens verursacht werden. In diesem Thema werden Verfahren beschrieben, die Ihnen bei der Behebung von Problemen beim Bootstrap-Prozess des AWS PCS-Knotens helfen. Weitere Informationen zur Fehlerbehebung beim Starten von EC2-Instances finden Sie unter Problembehandlung beim Starten von Amazon EC2 EC2-Instances im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Bootstrap-Fehler treten auf, wenn eine EC2-Instance erfolgreich gestartet wird, aber während des Beitritts zum PCS-Cluster fehlschlägt. AWS Der Bootstrap-Prozess umfasst zwei Hauptphasen:

- Knotenregistrierung — Die EC2-Instance ruft die RegisterComputeNodeGroupInstance AWS PCS-API-Aktion auf, um sich beim AWS PCS-Service zu registrieren. Fehler können aufgrund von Problemen in den folgenden Bereichen auftreten:
  - Berechtigungen
    - Falsches Instanzprofil
  - Netzwerk
    - Es kann keine Verbindung hergestellt werden mit AWS PCS-Endpunkte
    - Falsch konfiguriert AWS PCS-Endpunkt
    - Instanz in einem öffentlichen Subnetz ohne öffentliche IP
    - Multi-NIC Instanz in einem öffentlichen Subnetz
  - Geheimer Cluster
    - Der geheime Clusterschlüssel wurde gelöscht oder zum Löschen markiert
- Slurm-Integration — Die Instance läuft `slurmd` und tritt dem Slurm-Cluster bei. Fehler können aufgrund von Problemen in den folgenden Bereichen auftreten:
  - Berechtigungen
    - Sicherheitsgruppenkonfiguration
    - Slurmctld kann den Computerknoten nicht pingen
  - Benutzerdefiniertes AMI-Setup
    - Fehlende NVIDIA-Treiber
    - ResumeTimeout erreicht

Wie funktioniert Slurm AWS STK.

Es könnte Ihnen helfen, die Standardfunktion von Slurm mit der Funktionsweise von Slurm auf AWS PCS zu vergleichen.

Standardverarbeitung von Slurm-Jobs

Die folgenden Schritte finden bei der Standardverarbeitung von Slurm-Jobs statt:

1. Wenn Sie einen Job einreichen, `slurmctld` validiert er den Job und stellt ihn in eine Warteschlange.

2. Weist vorhandene Knoten zu, sobald Ressourcen verfügbar sind`slurmctld`.
3. `slurmd`Daemons führen Jobs auf zugewiesenen Knoten aus.

Verarbeitung von Slurm-Jobs aktiviert AWS STK.

Die folgenden Schritte finden bei der AWS PCS-Auftragsverarbeitung statt:

1. Wenn Sie einen Job einreichen, `slurmctld` validiert er den Job und stellt ihn in eine Warteschlange.
2. Wenn zusätzliche Kapazität benötigt wird, verwendet AWS PCS die Startvorlage für die Compute-Knotengruppe, um neue EC2-Instances zu starten.
3. Neue Instanzen starten per Bootstrap in den Cluster:
   a. Instanzen registrieren sich bei AWS PCS.
   b. Instanzen treten dem Slurm-Cluster bei.
4. Wenn die Ressourcen bereit sind, `slurmctld` weist sie Knoten zu (einschließlich der Knoten, die neu gebootet wurden).
5. `slurmd`Daemons führen Jobs auf zugewiesenen Knoten aus.

Instanzprotokolle abrufen

Der erste Schritt bei der Behebung von Bootstrap-Problemen mit Rechenknoten besteht darin, die Instanzprotokolle abzurufen. Sie können eine der folgenden Methoden verwenden:

AWS CLI

Rufen Sie mit dem folgenden Befehl die Konsolenausgabe vom Rechenknoten ab:

```
aws ec2 get-console-output --region us-east-1 --instance-id i-1234567890abcdef0 --
output text
```

*us-east-1*Ersetzen Sie es durch Ihre AWS Region und *i-1234567890abcdef0* durch Ihre Instance-ID.

AWS Systems Manager

Wenn Sie mit Systems Manager eine Verbindung zur Instanz herstellen können, können Sie die Bootstrap-Protokolldatei direkt anzeigen:

1. Stellen Sie mithilfe von Systems Manager eine Connect zur Instanz her. Weitere Informationen finden Sie unter Starten einer Sitzung im Systems Manager Manager-Benutzerhandbuch.
2. Sehen Sie sich die Bootstrap-Protokolldatei an:

```
sudo cat /var/log/amazon/pcs/bootstrap.log
```

> **Note**
>
> Wenn während der Initialisierungsphase ein Problem auftritt, müssen Sie möglicherweise etwa 20 Minuten warten, bevor Sie eine Verbindung mit der Instanz herstellen können. Systems Manager- und SSH-Dienste werden erst gestartet, nachdem die Initialisierung abgeschlossen ist oder wenn die Bootstrap-Ausführung im Fehlerfall ein Timeout erreicht.

Rufen Sie VPC/Subnet/Security Gruppen aus einer Instanz-ID ab

Um Probleme mit Ihren Rechenknoten zu beheben, müssen Sie möglicherweise Informationen über die VPC, das Subnetz und die Sicherheitsgruppen abrufen, die Ihren Instances zugeordnet sind. Wenn Sie Ihre Instanz-IDs nicht kennen, finden Sie weitere Informationen unter. Suchen nach Rechenknotengruppeninstanzen in AWS PCS

AWS-Managementkonsole

Um VPC-, Subnetz- und Sicherheitsgruppen abzurufen

1. Öffnen Sie die Amazon EC2-Konsole.
2. Wählen Sie Instances.
3. Wählen Sie in der Tabelle Instances die Instance-ID aus.
4. Suchen Sie die VPC-ID und die Subnetz-ID in der angezeigten Instanzübersicht für die Instance.
5. Wählen Sie in der Instanzübersicht die Registerkarte Sicherheit aus.
6. Suchen Sie die Sicherheitsgruppen auf der Registerkarte Sicherheit.

AWS CLI

Verwenden Sie den folgenden Befehl, um VPC-, Subnetz- und Sicherheitsgruppeninformationen für Ihre Instance abzurufen:

```
aws ec2 describe-instances --instance-ids i-1234567890abcdef0 --query
 'Reservations[*].Instances[*].
{InstanceId:InstanceId,VpcId:VpcId,SubnetId:SubnetId,SecurityGroups:SecurityGroups[*].GroupI
 --output table
```

Probleme bei der Knotenregistrierung

Die Knotenregistrierung ist die erste Aktion, die von einem Rechenknoten beim Bootstrap ausgeführt wird. Der Knoten ruft den AWS PCS-API-Endpunkt auf, um sich bei AWS PCS zu registrieren. Bei fehlgeschlagenen Registrierungen werden in der Regel Fehlermeldungen angezeigt, die den folgenden ähneln:

```
<13>Nov 13 16:23:50 user-data: [2025-11-13T16:23:50.510+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_init.sh: INFO: Registering node to cluster <clusterId>
<13>Nov 13 16:24:18 user-data: [2025-11-13T16:24:18.192+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_init.sh: INFO: Retriable exception detected.
<13>Nov 13 16:24:18 user-data: [2025-11-13T16:24:18.193+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_init.sh: INFO: Response is [specific error message]
<13>Nov 13 16:24:18 user-data: [2025-11-13T16:24:18.194+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_init.sh: INFO: Retrying in 31 seconds...
<13>Nov 13 16:24:18 user-data: [2025-11-13T16:24:18.192+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_init.sh: INFO: Retriable exception detected.
...
<13>Nov 13 16:25:18 user-data: [2025-11-13T16:25:18.195+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_init.sh: INFO: Registration timeout (600 seconds) reached. Exiting.
<13>Nov 13 16:25:18 user-data: [2025-11-13T16:25:18.200+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_init.sh: ERROR: Error: (2) occurred on line 1 when running /opt/aws/pcs/
bin/pcs_bootstrap_init.sh. Shutting down instance.
```

Falsches Instanzprofil

Wenn sich der Knoten aufgrund eines falschen Instanzprofils nicht registrieren kann, wird der folgende Fehler angezeigt:

```
<13>Nov 13 18:43:08 user-data: [2025-11-13T18:43:08.268+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_init.sh: INFO: Response is {
```

```
<13>Nov 13 18:43:08 user-data:   "__type":
 "com.amazon.coral.service#AccessDeniedException",
<13>Nov 13 18:43:08 user-data:   "Message": "User: arn:aws:sts::<accountId>:assumed-
role/<roleName>/<instanceId> is not authorized to perform:
 pcs:RegisterComputeNodeGroupInstance on resource:
 arn:aws:pcs:<regionCode>:<accountId>:cluster/<clusterId> as either the resource does
 not exist, some policy explicitly denies access, or no policy grants access",
<13>Nov 13 18:43:08 user-data:   "nodeID": null
<13>Nov 13 18:43:08 user-data: }
```

Stellen Sie sicher, dass das mit dem Compute-Knoten verknüpfte Instanzprofil über die `pcs:RegisterComputeNodeGroupInstance` entsprechende Berechtigung verfügt. Weitere Informationen zum Erstellen eines gültigen Instanzprofils finden Sie unterErstellen Sie ein Instanzprofil für AWS PCS.

Es kann keine Verbindung hergestellt werden mit AWS PCS-Endpunkte

Wenn sich Ihre Rechenknoten in einem privaten Subnetz befinden, stellen Sie sicher, dass Sie VPC-Endpunkte für AWS PCS konfiguriert haben oder dass Ihr Subnetz über eine Route zu einem NAT-Gateway für den Internetzugang verfügt. Weitere Informationen finden Sie hier:

- Greifen Sie über einen Schnittstellen-VPC-Endpunkt im Amazon Virtual Private Cloud AWS PrivateLink Cloud-Handbuch auf einen AWS Service zu.
- Endpunkte und Servicekontingenten für AWS STK..
- Connect Sie Ihre VPC mit anderen Netzwerken im Amazon Virtual Private Cloud Cloud-Benutzerhandbuch
- AWS PCS-Netzwerke

Falsch konfiguriert AWS PCS-Endpunkt

Wenn Sie eine Fehlermeldung ähnlich der folgenden sehen, überprüfen Sie die mit Ihrem AWS PCS-VPC-Endpunkt verknüpfte Richtlinie:

```
com.amazon.coral.security.AccessDeniedException: User: arn:aws:sts::xxx:assumed-
role/<roleName>/<instanceId> is not authorized to perform:
 pcs:RegisterComputeNodeGroupInstance on resource:
 arn:aws:pcs:<regionCode>:<accountId>:cluster/<clusterId> as either the resource does
 not exist, some policy explicitly denies access, or no policy grants access
```

Weitere Informationen zur Konfiguration von VPC-Schnittstellenendpunkten für AWS PCS finden Sie unter. Zugriff AWS Parallel Computing Service mithilfe eines Schnittstellenendpunkts (AWS PrivateLink)

Instanz in einem öffentlichen Subnetz ohne öffentliche IP

Wenn in Ihrem Subnetz die automatische Zuweisung öffentlicher IP-Adressen nicht aktiviert ist und Ihre Routenkonfiguration ein Internet-Gateway verwendet, können Instances nicht mit der AWS PCS-API kommunizieren.

Instanzen in einem Subnetz mit einem Internet-Gateway müssen eine öffentliche IP-Adresse haben. Wählen Sie eine der folgenden Optionen, um dieses Problem zu beheben:

- Fügen Sie Ihrer Cluster-VPC einen VPC-Endpunkt für AWS PCS hinzu. Dadurch können Instances mit AWS PCS kommunizieren, ohne dass eine öffentliche IP-Adresse das Internet-Gateway passieren muss.
- Verwenden Sie ein privates Subnetz mit einem NAT-Gateway, sodass keine öffentliche IP-Adresse erforderlich ist.
- Aktivieren Sie die automatische Zuweisung von öffentlichen IP-Adressen über Ihr Subnetz oder Ihre Startvorlage, sodass Instances die API über das Internet-Gateway kontaktieren können. Beachten Sie, dass diese Option nicht für Instances mit mehreren Netzwerkschnittstellen gilt.

Multi-NIC Instanz in einem öffentlichen Subnetz

Sie müssen ein privates Subnetz verwenden, wenn Sie einen Instance-Typ mit mehreren Netzwerkschnittstellen (NICs) verwenden.

AWS Öffentliche IP-Adressen können nur Instances zugewiesen werden, die mit einer einzigen Netzwerkschnittstelle gestartet wurden. Weitere Informationen zu IP-Adressen finden Sie unter Zuweisen einer öffentlichen IPv4-Adresse beim Instance-Start im Amazon EC2 EC2-Benutzerhandbuch für Linux-Instances.

Multi-NIC Instance-Typen benötigen ein NAT-Gateway oder einen internen Proxy im Subnetz, um auf den AWS PCS-Endpunkt zuzugreifen. Alternativ können Sie Ihrer Cluster-VPC einen VPC-Endpunkt für AWS PCS hinzufügen.

Der geheime Clusterschlüssel wurde gelöscht oder zum Löschen markiert

Wenn das Shared Secret von Slurm in AWS Secrets Manager gelöscht oder zum Löschen markiert wurde, können sich die Rechenknoten nicht registrieren und Ihr Cluster wird beeinträchtigt.

AWS PCS erstellt automatisch ein Shared Secret von Slurm in AWS Secrets Manager (mit dem Namensformat:`pcs!slurm-secret-<cluster-id>`), wenn Sie einen Cluster erstellen. Dieses Geheimnis ist für die sichere Kommunikation im Cluster erforderlich. Weitere Informationen finden Sie unter Arbeiten mit Clustergeheimnissen in AWS PCS.

Wenn dieses Geheimnis gelöscht oder zum Löschen markiert wird, können neue Knoten dem Cluster nicht beitreten, und der Controller oder andere Cluster-Daemons (wie `slurmd` und`slurmdbd`) können dem Cluster möglicherweise nicht wieder beitreten, wenn sie neu gestartet werden.

Um dieses Problem zu beheben, können Sie das gelöschte Geheimnis wiederherstellen, sofern es sich noch im Wiederherstellungsfenster befindet. Eine ausführliche Anleitung finden Sie unter Wiederherstellen eines AWS Secrets Manager Manager-Geheimnisses.

Wenn das Wiederherstellungsfenster abläuft, kann das Geheimnis nicht wiederhergestellt werden und der betroffene AWS PCS-Cluster kann nicht wiederhergestellt werden. Sie müssen einen neuen Cluster mit derselben Konfiguration erstellen. AWS PCS erstellt automatisch ein neues Scheduler-Secret.

Probleme beim Zusammenschluss mit Slurm-Clustern

Nach erfolgreicher Knotenregistrierung versucht der Rechenknoten, dem Slurm-Cluster beizutreten. Der `slurmd` Daemon auf dem Knoten kontaktiert den Slurm-Controller, um sich beim Cluster zu registrieren. Bei Fehlern beim Slurm-Join werden normalerweise Fehlermeldungen angezeigt, die den folgenden ähneln:

```
<13>Nov  5 17:20:29 user-data: [2024-11-05T17:20:28+00:00] FATAL:
 Mixlib::ShellOut::ShellCommandFailed: service[slurmd] (aws-pcs-slurm::finalize_slurm
 line 18) had an error: Mixlib::ShellOut::ShellCommandFailed: Expected process to exit
 with [0], but received '1'
<13>Nov  5 17:20:29 user-data: ---- Begin output of ["/usr/bin/systemctl", "--system",
 "start", "slurmd"] ----
<13>Nov  5 17:20:29 user-data: STDOUT:
<13>Nov  5 17:20:29 user-data: STDERR: Job for slurmd.service failed because the
 control process exited with error code. See "systemctl status slurmd.service" and
 "journalctl -xe" for details.
```

```
<13>Nov  5 17:20:29 user-data: ---- End output of ["/usr/bin/systemctl", "--system",
 "start", "slurmd"] ----
```

Sicherheitsgruppenkonfiguration

Stellen Sie sicher, dass Ihre Sicherheitsgruppen korrekt konfiguriert sind, um die Kommunikation zwischen Rechenknoten und dem Slurm-Controller zu ermöglichen. Die Sicherheitsgruppen müssen den folgenden Verkehr zulassen:

- Port 6817 für `slurmd` die Kommunikation mit `slurmctld`
- Port 6818 zum Pingen `slurmctld` `slurmd`

Weitere Informationen zu den Anforderungen an Sicherheitsgruppen finden Sie in den folgenden Themen:

- Sicherheitsgruppen für AWS PCS erstellen
- Startvorlagen für AWS PCS erstellen
- Anforderungen und Überlegungen zur Sicherheitsgruppe

> **Important**
>
> Die Cluster-Sicherheitsgruppe, die Sie Ihrem Cluster bei der Clustererstellung zugeordnet haben, muss auch in den Sicherheitsgruppen Ihrer Compute-Knotengruppe konfiguriert werden, damit Rechenknoten mit dem Controller kommunizieren können.

Fehlende NVIDIA-Treiber

Wenn die Instanz korrekt bootet, Jobs aber nicht gestartet werden und Sie in Ihren Instanzprotokollen Fehlermeldungen wie die folgenden sehen, fehlen Ihnen möglicherweise NVIDIA-Treiber:

```
<13>Dec  2 13:52:00 user-data: [2024-12-02T13:52:00.094+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_config_always.sh: INFO: nvidia-smi not found!
...
<13>Dec  2 13:54:10 user-data: Job for slurmd.service failed because the control
 process exited with error code. See "systemctl status slurmd.service" and "journalctl
 -xe" for details.
```

```
<13>Dec  2 13:54:12 user-data: [2024-12-02T13:54:12.718+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_finalize.sh: INFO: systemctl could not start slurmd!
```

Wenn Sie eine Verbindung mit der Instance herstellen und den `slurmd` Daemon-Status überprüfen, wird möglicherweise ein Fehler ähnlich dem folgenden angezeigt:

```
$ systemctl status slurmd
...
fatal: can't stat gres.conf file /dev/nvidia0: No such file or directory
```

Um dieses Problem zu beheben, installieren Sie NVIDIA-Treiber auf Ihrem benutzerdefinierten AMI. Weitere Informationen finden Sie unter Schritt 4 — (Optional) Zusätzliche Treiber, Bibliotheken und Anwendungssoftware installieren.

ResumeTimeout erreicht

Wenn ein Rechenknoten und seine EC2-Instance beendet werden, weil der Knoten fehlerhaft ist, unterstützt AWS PCS das AMI möglicherweise nicht, oder es liegen Netzwerkprobleme vor. Die EC2-Instance läuft ungefähr 30 Minuten lang, bis der Slurm-Wert erreicht ResumeTimeout ist, und markiert den Knoten als. `DOWN`

Wenn die Instance nicht korrekt bootet und nicht bei AWS PCS registriert ist (kein `RegisterComputeNodeGroupInstance` Aufruf für die EC2-Instance), überprüfen Sie Ihre Instance-Logs auf Fehlermeldungen, die den folgenden ähneln:

```
/opt/aws/pcs/bin/pcs_bootstrap_init.sh: No such file or directory
```

Dieser Fehler weist darauf hin, dass die AWS PCS-Bootstrap-Software nicht Teil des AMI ist. Um dieses Problem zu beheben, stellen Sie sicher, dass Ihr benutzerdefiniertes AMI die AWS PCS-Bootstrap-Software enthält. Weitere Informationen finden Sie unter Benutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS.

Slurmctld kann den Computerknoten nicht pingen

Wenn die Instanz die Bootstrap-Prozedur korrekt ausführt und bei AWS PCS registriert ist, sie aber `slurmctld` nicht sehen und keine Jobs an sie senden kann, wird die Instanz `DOWN` nach einiger Zeit auf eingestellt und dann beendet.

Dies kann durch falsch konfigurierte Sicherheitsgruppen verursacht werden. Zum Beispiel, wenn Port 6817 für die Kommunikation mit aktiviert ist`slurmctld`, aber Port 6818 fehlt, `slurmd` um Ping zu ermöglichen`slurmctld`. `slurmd`

Stellen Sie sicher, dass Ihre Sicherheitsgruppen alle erforderlichen Regeln enthalten, wie unter beschrieben. Anforderungen und Überlegungen zur Sicherheitsgruppe

Behebung von Fehlern bei der Einreichung von Jobs aufgrund eines MaxJobCount Limits

Problem: Die Jobübermittlung schlägt fehl und die folgende Fehlermeldung wird angezeigt:

```
sbatch: error: Slurm temporarily unable to accept job, sleeping and retrying
```

Dieser Fehler tritt auch dann auf, wenn die Anzahl der laufenden und ausstehenden Jobs deutlich unter dem Joblimit des Clusters zu liegen scheint.

Ursache: Das `MaxJobCount` Limit umfasst alle Jobs, die von Slurm verfolgt werden, nicht nur laufende oder ausstehende Jobs. Abgeschlossene Jobs bleiben für einen bestimmten Zeitraum (standardmäßig 5 Minuten) im Speicher von Slurm, bevor sie gelöscht werden. In Zeiten mit hohem Auftragsdurchsatz kann die Gesamtzahl der aktiven und kürzlich abgeschlossenen Jobs das Limit überschreiten.

Sie können die Gesamtzahl der Jobs überprüfen, indem Sie den folgenden Befehl auf einem Clusterknoten ausführen:

```
scontrol show jobs | grep -c JobId
```

Dies zeigt die Gesamtzahl der Jobs, die Slurm verfolgt, einschließlich abgeschlossener Jobs, die noch gelöscht werden müssen.

Lösung: Ziehen Sie einen der folgenden Ansätze in Betracht:

- Einen größeren Cluster erstellen — Wenn Ihr Workload durchweg mehr gleichzeitige Jobs erfordert, erstellen Sie einen neuen Cluster mit einer größeren Größe. Weitere Informationen zu Clustergrößen und deren Beschränkungen finden Sie unterClustergröße in AWS PCS.
- Reduzieren Sie die Anzahl der eingereichten Jobs — Passen Sie Ihre Skripte für die Einreichung von Jobs an, um Jobs langsamer einzureichen, sodass die Zeit für erledigte Jobs aus dem Tracking von Slurm gelöscht wird.

Dokumentenhistorie für die AWS PCS-Benutzerhandbuch

In der folgenden Tabelle werden die wichtigen Änderungen an der Dokumentation für AWS PCS beschrieben.

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
01. Juni 2026	Neues Thema: PCS-ready DLAMI	Dokumentation für das AWS PCS-ready DLAMI Base GPU AMI hinzugefügt, ein AWS-maintained AMI zum Ausführen AI/ML und HPC-Workloads auf PCS. AWS Weitere Informationen finden Sie unter Verwenden von PCS-ready DLAMI mit AWS STK..	N/A
28. Mai 2026	Neue Funktion: Berechnung der Leerlaufzeit beim Herunterskalieren auf Knotengruppenebene	Einzelne Compute-Knotengruppen können jetzt die Leerlaufzeit beim Herunterskalieren auf Clusterebene mit ihren eigenen Einstellungen überschreiben. `scaleDownIdleTimeInSeconds` Für diese Funktion ist Slurm Version 25.11 oder höher erforderlich.	N/A
26. Mai 2026	AWS PCS wurde im asiatisch-pazifischen Raum (Osaka) und in	AWS PCS ist jetzt im asiatisch-pazifischen Raum (Osaka) (ap-northeast-3) und in Südamerik	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
	Südamerika (São Paulo) veröffentlicht	a (São Paulo) (sa-east-1) erhältlich. CloudFormation Vorlagen sind für den Einstieg im asiatisch-pazifisc hen Raum (Osaka) und in Südamerika (São Paulo) verfügbar. AWS-Regionen Weitere Informationen erhalten Sie unter Verwenden Sie CloudFormation um ein Beispiel zu erstellen AWS PCS-Cluster und CloudFormation Vorlagen zum Erstellen eines Musters AWS PCS-Clust er.	

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
18. Mai 2026	Neue Funktion: Gemischte Skalierung skapazität für Rechenknotengruppen	Sie können jetzt Rechenknotengruppen mit gemischter Skalierung konfigurieren, wobei die Mindestanzahl der Instanzen größer als Null und weniger als die maximale Anzahl an Instanzen ist. Dadurch wird bei dynamischer Skalierung ein Basiswert an ständig verfügbarer Kapazität aufrechterhalten. Für diese Funktion ist 24.05 oder höher erforderlichSlurm. Weitere Informationen finden Sie unter Erstellen einer Compute-Knotengruppe in AWS STK..	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
7. Mai 2026	Der PCS-Agent wurde aktualisiert	Das AMI-Thema für AWS PCS Agent 1.4.0-1 wurde aktualisiert. Der AWS PCS-Agent unterstützt jetzt die Konfiguration und den Start des `slurmd` Daemons für jede Slurm-Version, die mit dem Slurm-Controller kompatibel ist, der auf dem Cluster läuft. Weitere Informationen erhalten Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS und AWS Versionen von PCS-Agenten.	N/A
29. April 2026	Das Slurm-25.05-Installationsprogramm wurde aktualisiert	Das AMI-Thema für den Slurm-Installer wurde am 25.05.7-1 aktualisiert. Weitere Informationen erhalten Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS und AWS Versionen von PCS-Agenten.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
23. April 2026	AWS PCS in Europa (Spanien) veröffentlicht	AWS PCS ist jetzt in Europa (Spanien) (eu-south-2) verfügbar. CloudFormation Vorlagen sind für den Einstieg in Europa (Spanien) verfügbar AWS-Regio n. Weitere Informati onen erhalten Sie unter Verwenden Sie CloudFormation um ein Beispiel zu erstellen AWS PCS-Cluster und CloudFormation Vorlagen zum Erstellen eines Musters AWS PCS-Clust er.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
23. April 2026	Für Slurm 25.11 aktualisiert	Das Benutzerhandbuch für die Unterstützung von Slurm 25.11.2-1 wurde aktualisiert. Slurm 25.11 ist jetzt die Standardversion. Beispiel-AMIs für Slurm 25.11 basieren jetzt auf Amazon Linux 2023. Weitere Informationen finden Sie hier: • Slurm-Versionen in AWS STK. • Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS • Versionshinweise für AWS PCS-Beispiel-AMIs	N/A
22. April 2026	Neue Funktion: Scheduler-Auditprotokolle	AWS PCS liefert Scheduler-Audit-Logs jetzt separat über den `PCS_SCHEDULER_AUDIT_LOGS` Log-Typ für Cluster, auf denen Slurm 25.11 oder höher ausgeführt wird. Weitere Informationen finden Sie unter Scheduler-Auditprotokolle in AWS PCS.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
22. April 2026	Die Scheduler-Logs wurden mit slurmdbd und slurmrestd aktualisiert	AWS PCS liefert jetzt `slurmdbd` Logs (Slurm 24.11+) und `slurmrestd` Logs (Slurm 25.05+) über den vorhanden en Logtyp. `PCS_SCHEDULER_LOGS` Weitere Informationen finden Sie unter Scheduler loggt sich in AWS PCS ein.	N/A
16. April 2026	AWS PCS in Europa (Mailand) veröffentlicht	AWS PCS ist jetzt in Europa (Mailand) (eu-south-1) verfügbar. CloudFormation Vorlagen sind für den Einstieg in Europa (Mailand) verfügbar. AWS-Regio n Weitere Informati onen erhalten Sie unter Verwenden Sie CloudFormation um ein Beispiel zu erstellen AWS PCS-Cluster und CloudFormation Vorlagen zum Erstellen eines Musters AWS PCS-Clust er.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
10. März 2026	Der PCS-Agent wurde aktualisiert	Das AMI-Thema für den AWS PCS-Agenten 1.3.2-1 wurde aktualisiert. Ein Problem mit Auswirkungen auf RHEL 8.10 und Rocky Linux 8.10 Compute Node Bootstrap wurde behoben. Weitere Informationen erhalten Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS und AWS Versionen von PCS-Agenten.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
11. Februar 2026	AWS PCS wurde im asiatisch-pazifischen Raum (Mumbai) und in Europa (Paris) veröffentlicht	AWS PCS ist jetzt in Asien-Pazifik (Mumbai) (ap-south-1) und Europa (Paris) (eu-west-3) verfügbar. CloudFormation Vorlagen sind für den Einstieg im asiatisch-pazifischen Raum (Mumbai) AWS-Region und Europa (Paris) verfügbar. AWS-Region Weitere Informationen erhalten Sie unter Verwenden Sie CloudFormation um ein Beispiel zu erstellen AWS PCS-Cluster und CloudFormation Vorlagen zum Erstellen eines Musters AWS PCS-Cluster.	N/A
18. November 2025	Neue Funktion: Slurm REST API	Die Slurm-REST-API wird jetzt für Slurm 25.05 oder höher unterstützt. Weitere Informationen finden Sie unter Schlurm-REST-API ein AWS STK..	AWS-SDK: 18.11.2025

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
17. November 2025	Die Slurm-Installationsprogramme wurden aktualisiert	Das AMI-Thema für die AWS PCS Slurm-Installer 24.11.7-1 und 25.05.5-1 wurde aktualisiert. Weitere Informationen erhalten Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS und AWS Versionen von PCS-Agenten.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
10. November 2025	Neue Funktion: Unterstützung für das Slurm-CLI-Filter-Plugin	AWS PCS unterstützt jetzt Slurm-CLI-Filter-Plugins, um benutzerdefinierte Lua-Skripte auszuführen, die die Parameter für die Auftragsübermittlung validieren und ändern, bevor sie den Slurm-Controller erreichen. Verwenden Sie CLI-Filter, um benutzerdefinierte Richtlinien durchzusetzen, Standardparameter festzulegen und Benutzerführung bei der Einreichung von Jobs bereitzustellen. Für diese Funktion ist Slurm Version 25.05 oder höher erforderlich. Weitere Informationen finden Sie unter Verwenden Sie die Slurm CLI Filter Plugins, um die Einreichung von Jobs anzupassen in AWS STK..	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
7. November 2025	Der PCS-Agent wurde aktualisiert	Das AMI-Thema für den AWS PCS-Agenten 1.3.1-1 wurde aktualisiert. Weitere Informationen erhalten Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS und AWS Versionen von PCS-Agenten.	N/A
3. November 2025	Die Installer für PCS Agent und Slurm wurden aktualisiert	Das AMI-Thema für den AWS PCS-Agent 1.3.0-1 und die Slurm-Installer 24.11.6-2, 24.05.8-2 und 23.11.10-4 wurde aktualisiert. Die Liste der unterstützten Betriebssysteme wurde aktualisiert. Weitere Informationen erhalten Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS und AWS Versionen von PCS-Agenten.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
23. Oktober 2025	Aktualisierter Inhalt: pcs-multi-cluster-login-configure.sh	Einige Fehler im Konfigurationsskript für den Multi-Cluster-Login-Knoten wurden behoben. Weitere Informationen finden Sie unter AWS Skriptcode für die Konfiguration des PCS-Multi-Cluster-Anmeldeknotens.	N/A
21. Oktober 2025	Neue Funktion: Geheime Cluster-Rotation	AWS PCS unterstützt jetzt die Rotation von Cluster-Secrets, um die Sicherheit zu erhöhen. Weitere Informationen finden Sie unter Rotierende Clustergeheimnisse in AWS PCS. Die Mindestadministratorrechte wurden aktualisiert, um die geheime Cluster-Rotation zu unterstützen. Weitere Informationen finden Sie unter Mindestberechtigungen für AWS STK..	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
17. Oktober 2025	Neues Thema: Konfigurationsskript für Multi-Cluster-Login-Knoten	Es wurde ein neues Thema hinzugefügt, das ein Skript zur Konfiguration eines eigenständigen Anmeldeknotens für die Verbindung mit mehreren AWS PCS-Clustern bereitstellt. Das Skript automatisiert die Konfiguration mehrerer `sackd` Slurm-Daemons und erstellt Aktivierungsskripten für die Cluster-Interaktion. Weitere Informationen finden Sie unter Einen eigenständigen Anmeldeknoten mit mehreren Clustern in AWS PCS verbinden.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
16. Oktober 2025	Für Slurm 25.05 aktualisiert	Das Benutzerhandbuch für die Unterstützung von Slurm 25.05 wurde aktualisiert. Slurm 25.05 ist jetzt die Standardversion. Weitere Informationen finden Sie hier: • Slurm-Versionen in AWS STK. • Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS • Versionshinweise für AWS PCS-Beispiel-AMIs	N/A
16. Oktober 2025	Der PCS-Agent wurde aktualisiert	Das AMI-Thema für AWS PCS Agent 1.2.2-1 wurde aktualisiert. Weitere Informationen erhalten Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS und AWS Versionen von PCS-Agenten.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
2. Oktober 2025	Neue Funktionen: Neustart des Slurm-Knotens, Cluster-Updates und benutzerdefinierte Slurm-Einstellungen	AWS PCS bietet Unterstützung für mehrere neue Funktionen: • Neustart des Slurm-Knotens — Verwenden Sie den nativen `scontrol reboot` Befehl von Slurm, um Rechenknoten ohne Instanzersatz neu zu starten. Weitere Informationen finden Sie unter Compute-Knoten mit eingescha ltetem Slurm neu starten AWS STK.. • Cluster-Updates — Ändern Sie Cluster-Konfigurationen nach der Erstellung ohne Neuerstellungen. Weitere Informationen finden Sie unter Aktualisierung eines Clusters in AWS PCS. • Benutzerdefinierte Slurm-Einstellungen — Konfigurieren Sie erweiterte Slurm-Parameter für Cluster-, Queue- und Compute	2025-10-01

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
		Node Group-Ressourcen. Weitere Informationen finden Sie unter Konfiguration benutzerdefinierter Slurm-Einstellungen in AWS STK..	
23. September 2025	Neues Thema zur Fehlerbehebung: Bootstrap-Probleme mit Compute-Knoten	Es wurde eine Anleitung zur Fehlerbehebung für die Diagnose und Lösung von Bootstrap-Problemen mit Compute-Knoten hinzugefügt. Weitere Informationen finden Sie unter Beheben Sie Probleme mit dem Bootstrap und der Registrierung von Rechenknoten in AWS STK..	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
17. September 2025	Neue Funktion: Kapazitätsblöcke für ML	AWS PCS unterstützt jetzt Amazon EC2 Capacity Blocks for ML, mit denen Sie GPU-based beschleunigte Recheninstanzen für Ihre Cluster reservieren können. Weitere Informationen finden Sie unter Verwenden von Amazon EC2 EC2-Kapazitätsblöcken für ML mit AWS PCS. Mindestberechtigungen zur Unterstützung von Capacity Blocks gehören jetzt zu den Mindestberechtigungen für einen Service-Administrator. Weitere Informationen finden Sie unter Mindestberechtigungen für AWS STK..	2025-09-17
11. Septembe 2025	Aktualisierung der verwalteten AWS-Richtlinien	AWS PCS hat das aktualisiert AWSPCSServiceRolePolicy , um Capacity Blocks zu unterstützen. Weitere Informationen finden Sie unter AWS verwaltete Richtlinien für AWS Dienst für parallele Datenverarbeitung.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
14. August 2(	Die Dokumentation zum Instanzprofil wurde aktualisiert	Die Dokumentation zu Instanzprofilen wurde um umfassende CLI-Anweisungen zum Erstellen von IAM-Rolle n und Instanzprofilen erweitert. Es wurden schrittweise Anleitung en zum Einrichten von Instanzprofilen mithilfe von hinzugefügt AWS CLI und die Anleitung zur Suche nach Instanzpr ofilen, die mit AWS PCS verwendet werden, verbessert. Weitere Informationen finden Sie unter IAM-Insta nzprofile für AWS Dienst für parallele Datenvera rbeitung.	2025-08-14

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
1. August 20	Neues Thema: SPANK-Plugins	Es wurde eine Dokumentation für SPANK-Plugins (Slurm Plug-in Architecture for Node and Job Kontrol) hinzugefügt, mit der Sie das Verhalten von Slurm beim Start und der Ausführung von Jobs auf PCS-Clustern erweitern und ändern können. AWS Weitere Informationen finden Sie unter Erweitern Sie die Slurm-Funktionalität auf AWS PCS mit SPANK-Plugins.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
1. August 202	Unterstützung für IPv6-Netzwerke	Unterstützung für IPv6-Netzwerke bei der Erstellung von AWS PCS-Clustern hinzugefügt. Sie können jetzt IPv6 als Netzwerktyp für Ihren Cluster auswählen, mit entsprechenden Aktualisierungen der VPC-Anforderungen, der Subnetzkonfiguration, der Sicherheitsgruppeneinstellungen und der Verfahren zur Clustererstellung. Weitere Informationen erhalten Sie unter AWS Anforderungen und Überlegungen zu PCS VPC und Subnetzen und Einen Cluster erstellen in AWS STK..	2025-08-01

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
3. Juli 2025	AWS PCS in Europa (London) veröffentlicht	AWS PCS ist jetzt in Europa (London) (eu-west-2) verfügbar. CloudFormation Vorlagen sind für den Einstieg in Europa (London) verfügbar. AWS-Region Weitere Informationen erhalten Sie unter Verwenden Sie CloudFormation um ein Beispiel zu erstellen AWS PCS-Cluster und CloudFormation Vorlagen zum Erstellen eines Musters AWS PCS-Cluster.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
1. Juli 2025	Die Anweisungen für die Konsole wurden aktualisiert	Sie können jetzt AWS PCS ein grundlegendes Instanzprofil und eine Sicherheitsgruppe für Sie erstellen lassen, wenn Sie einen Cluster und eine Rechenknotengruppe in der Konsole erstellen. Weitere Informationen finden Sie unter: • Einen Cluster erstellen in AWS STK. • Erstellen einer Compute-Knotengruppe in AWS STK. • IAM-Instanzprofile für AWS Dienst für parallele Datenverarbeitung	N/A
23. Juni 2025	Neue verwaltete Richtlinie: AWSPCSComputeNodePolicy	Es wurde eine neue verwaltete Richtlinie hinzugefügt, die AWS PCS-Rechenknoten die Erlaubnis erteilt, sich mit AWS PCS-Clustern zu verbinden. Weitere Informationen finden Sie unter AWS verwaltete Richtlinie: AWSPCSComputeNodePolicy.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
19. Juni 2025	Neues Thema: Protokolle zum Abschluss von Jobs	Verwenden Sie Protokolle zum Abschluss von Aufträgen, um ohne zusätzliche Kosten Details zu den abgeschlossenen Aufträgen aufzuzeichnen. Weitere Informationen finden Sie unter Auftragsabschlussprotokolle in AWS PCS.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
18. Juni 2025	AWS PCS-Veröffentlichung in AWS GovCloud (US)	AWS PCS ist jetzt in AWS GovCloud (US-East) (us-gov-east-1) und AWS GovCloud () (us-gov-west-1US-West) verfügbar. CloudFormation Vorlagen sind für den Einstieg in der verfügbar. AWS GovCloud (US) Regions Weitere Informationen erhalten Sie unter Verwenden Sie CloudFormation um ein Beispiel zu erstellen AWS PCS-Cluster und CloudFormation Vorlagen zum Erstellen eines Musters AWS PCS-Cluster. Weitere Informationen zu den AWS PCS-Diensendpunkten finden Sie AWS GovCloud (US) Regions unterEndpunkte und Servicekontingenten für AWS STK.. Weitere Informationen zu den Unterschieden in AWS GovCloud (US) Regions finden	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
		Sie unter AWS PCS AWS GovCloud (US) im AWS GovCloud (US) Benutzerhandbuch.	
18. Juni 2025	Der PCS-Agent wurde aktualisiert	Das AMI-Thema für AWS PCS Agent 1.2.1-1 wurde aktualisiert. Weitere Informationen finden Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS.	N/A
15. Mai 2025	Neue Funktion: Buchhaltung	Die Slurm-Buchhaltung wird jetzt für Slurm 24.11 oder höher unterstützt. Weitere Informationen finden Sie unter Slurm-Buchhaltung in AWS STK..	AWS-SDK: 15.05.2025

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
15. Mai 2025	Für Slurm 24.11 aktualisiert	Das Benutzerhandbuch für die Unterstützung von Slurm 24.11.5 wurde aktualisiert. Weitere Informationen finden Sie hier: • Slurm-Versionen in AWS STK. • Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS • Versionshinweise für AWS PCS-Beispiel-AMIs	N/A
5. Mai 2025	Häufig gestellte Fragen zu Slurm-Versionen aktualisiert	Die häufig gestellten Fragen (FAQ) zu Slurm-Versionen, die kurz vor dem Ende des Lebenszyklus (EOL) stehen, wurden aktualisiert. Weitere Informationen finden Sie unter Häufig gestellte Fragen zu Slurm-Versionen in AWS STK..	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
17. April 2025	Neues Thema: Wie erhalte ich Informationen zu Compute-Knotengruppen	Erfahren Sie, wie Sie Details für eine AWS PCS-Compute-Knotengruppe abrufen, z. B. ihre ID, ARN und AMI-ID. Weitere Informationen finden Sie unter Details zur Compute-Knotengruppe in AWS PCS abrufen.	N/A
2. April 2025	Das Slurm-Installationsprogramm wurde aktualisiert	Das AMI-Thema für den Slurm-Installer wurde am 24.05.7-1 aktualisiert. Weitere Informationen finden Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS.	N/A
28. März 2025	Kontingente für die maximale Anzahl von Rechenknotengruppen und Warteschlangen hinzugefügt	Interne, nicht einstellbare Kontingente für die maximale Anzahl von Rechenknotengruppen pro Cluster und die maximale Anzahl von Warteschlangen pro Cluster wurden hinzugefügt. Weitere Informationen finden Sie unter Interne Kontingente.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
14. März 2025	Ein Eigenschaftsschlüssel in der CloudFormation Vorlage wurde geändert	`Id` ist jetzt `TemplateId` für die `CustomLaunchTemplate` Eigenschaft in der CloudFormation Vorlage. Weitere Informationen finden Sie unter Ressourcen in Teile einer CloudFormation Vorlage für AWS STK..	N/A
13. März 202	Versionsinformationen für den AWS PCS-Agenten und Slurm hinzugefügt	Es wurde ein neues Thema hinzugefügt, das die Änderungen für jede Version des AWS PCS-Agenten beschreibt. Weitere Informationen finden Sie unter AWS Versionen von PCS-Agenten. Dem Thema Slurm-Versionen wurden weitere Informationen hinzugefügt, in denen wichtige Support-Daten und detaillierte Versionshinweise für die AWS PCS-Unterstützung für Slurm beschrieben werden. Weitere Informationen finden Sie unter Slurm-Versionen in AWS STK..	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
7. März 2025	Der PCS-Agent wurde aktualisiert	Das AMI-Thema für AWS PCS Agent 1.2.0-1 wurde aktualisiert. Weitere Informationen finden Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS.	N/A
3. Februar 2025	Es wurde ein Thema zur Verwendung AWS CloudFormation mit AWS PCS hinzugefügt	Dem Benutzerhandbuch wurde ein Thema hinzugefügt, das ein Beispiel für die Verwendung CloudFormation mit AWS PCS enthält. Das Thema enthält ein Verfahren zur Verwendung einer CloudFormation Beispielvorlage zur Erstellung des AWS PCS-Beispielclusters und eine kurze Beschreibung der Abschnitte dieser Vorlage. Weitere Informationen finden Sie unter Erste Schritte mit CloudFormation AWS PCS.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
18. Dezembe	Für Slurm 24.05 aktualisiert	Das Benutzerhandbuch für die Unterstützung von Slurm 24.05 wurde aktualisiert. Weitere Informationen erhalten Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS und Versionshinweise für AWS PCS-Beispiel-AMIs.	N/A
18. Dezembe	Die NVIDIA-Versionen für Slurm 23.11-Beispiel-AMIs wurden aktualisiert	Die NVIDIA-Treiber- und CUDA-Versionen in den Slurm 23.11-Beispiel-AMIs wurden aktualisiert. Weitere Informationen finden Sie unter Versionshinweise für AWS PCS-Beispiel-AMIs.	N/A
17. Dezembe	Das Slurm-Installationsprogramm wurde aktualisiert	Das AMI-Thema für den Slurm-Installer 23.11.10-3 wurde aktualisiert. Weitere Informationen finden Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
13. Dezember 2024	Der PCS-Agent wurde aktualisiert	Das AMI-Thema für den AWS PCS-Agenten 1.1.1-1 wurde aktualisiert. Weitere Informationen finden Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS.	N/A
6. Dezember 2024	Der PCS-Agent und das Slurm-Installationsprogramm wurden aktualisiert	Das AMI-Thema für den AWS PCS-Agent 1.1.0-1 und den Slurm-Installer 23.11.10-2 wurde aktualisiert. Weitere Informationen finden Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS.	N/A
6. Dezember 2024	Ein Thema zur Betriebssystemunterstützung wurde hinzugefügt	Weitere Informationen finden Sie unter Unterstützte Betriebssysteme in AWS PCS.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
8. November	Das Benutzerhandbuch wurde neu organisiert	Wir haben das Benutzerhandbuch neu organisiert, um die Themen auf die oberste Ebene zu bringen, einige Themen auf eigene Seiten verschoben und ähnliche Themen gruppiert.	N/A
7. November	Aktualisierte AMI-Themen	Das AMI-Thema für Slurm 23.11.10 und libjwt 17.0 wurde aktualisiert. Weitere Informationen erhalten Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS und Schritt 3 — Slurm installieren. Die Versionshinweise für AMIs wurden vereinfacht und korrigiert. Weitere Informationen finden Sie unter Versionshinweise für AWS PCS-Beispiel-AMIs.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
7. November	Es wurde ein neues Thema zur Verwendung verschlüsselter EBS-Volumes mit AWS PCS hinzugefügt	Es wurde ein Thema hinzugefügt, das die KMS-Schlüsselrichtlinie beschreibt, die für verschlüsselte EBS-Volumes in AWS PCS erforderlich ist. Weitere Informationen finden Sie unter Erforderliche KMS-Schlüsselrichtlinie für die Verwendung mit verschlüsselten EBS-Volumes in AWS STK..	N/A
18. Oktober 2024	AWS PCS Agent 1.0.1-1 veröffentlicht	Die AMI-related Dokumentation wurde aktualisiert und bezieht sich nun auf die AWS PCS-Agent-Version 1.0.1-1. Weitere Informationen erhalten Sie unter Softwareinstallationsprogramme zum Erstellen benutzerdefinierter AMIs für AWS PCS und Schritt 2 — Installieren Sie den AWS PCS-Agenten.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
10. Oktober 2	Ein Kapitel zur Problembehandlung wurde hinzugefügt	Es wurde ein Kapitel zur Fehlerbehebung hinzugefügt, in dem es um das automatische Ersetzen von EC2-Instances nach einem Neustart geht. Weitere Informationen finden Sie unter Behebung von Problemen in AWS Dienst für parallele Datenverarbeitung.	N/A
23. Septembe 2024	Die Mindestberechtigungen für die Verwendung von API-Aktionen und für einen Dienstadministrator wurden aktualisiert	Die `ec2:DescribeInstanceTypeOfferings` Erlaubnis ist jetzt für die `CreateComputeNodeGroup` und `UpdateComputeNodeGroup` API-Aktionen erforderlich. Weitere Informationen finden Sie unter Mindestberechtigungen für AWS STK..	N/A
5. Septembe 2024	Die Beispiel-IAM-Richtlinie für die Mindestberechtigungen für einen Dienstadministrator wurde aktualisiert	Weitere Informationen finden Sie unter Mindestberechtigungen für einen Dienstadministrator.	N/A

Date	Änderungen	Aktualisierungen der Dokumentation	API-Versionen wurden aktualisiert
5. Septembe 2024	Dem JSON wurde auf der Seite mit verwalteten Richtlinien eine fehlende Berechtigung hinzugefügt	Dies war nur eine Korrektur der Dokumentation. Die tatsächlich verwaltete Richtlinie wurde nicht geändert. Weitere Informationen finden Sie unter AWS verwaltete Richtlinien für AWS Dienst für parallele Datenverarbeitung.	N/A
28. August 2(	Seite „Verwaltete Richtlinien“ hinzugefügt	Weitere Informationen finden Sie unter AWS verwaltete Richtlinien für AWS Dienst für parallele Datenverarbeitung.	N/A
28. August 2(	AWS PCS-Version	Erste Version des AWS PCS-Benutzerhandbuchs.	AWS SDK: 2024-08-28

AWS Glossar

Die neueste AWS Terminologie finden Sie im AWS Glossar in der AWS-Glossar Referenz.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.