aws

User Guide

AWS PCS

AWS PCS: User Guide

Table of Contents

Was ist AWS Parallel Computing Service?

AWS Parallel Computing Service (AWS PCS) ist ein verwalteter Service, der es einfacher macht, High Performance Computing (HPC) -Workloads auszuführen und zu skalieren und wissenschaftliche und technische Modelle für die AWS Verwendung von Slurm zu erstellen. Wird AWS PCS zum Aufbau von Rechenclustern verwendet, die erstklassige AWS Rechenleistung, Speicherung, Netzwerke und Visualisierung 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.

Die wichtigsten Konzepte

Ein Cluster 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 verwenden, 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 welchem die VPC Cluster-Ressourcen gestartet werden sollen. Der Scheduler akzeptiert und plant Jobs und startet auch die Rechenknoten (EC2Instanzen), die diese Jobs verarbeiten.

Compute-Knotengruppe

Eine Rechenknotengruppe ist eine Sammlung von Rechenknoten, die AWS PCS verwendet werden, um Jobs auszuführen oder interaktiven Zugriff auf einen Cluster zu ermöglichen. Wenn Sie eine Rechenknotengruppe definieren, geben Sie allgemeine Merkmale wie EC2 Amazon-Instance-Typen, minimale und maximale Instance-Anzahl, VPC Zielsubnetze, Amazon Machine Image (AMI), Kaufoption und benutzerdefinierte Startkonfiguration an. AWS PCSverwendet 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 verbleibt in der Warteschlange, bis AWS PCS er für die Ausführung auf einer Rechenknotengruppe geplant ist. 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 senden Jobs nicht direkt an einen Rechenknoten oder eine Rechenknotengruppe.

Systemadministrator

Ein Systemadministrator stellt einen Cluster bereit, verwaltet und betreibt ihn. Sie können AWS PCS über AWS Management Console AWS PCSAPI, und AWS SDK darauf zugreifen. Sie haben über SSH oder Zugriff auf bestimmte Cluster AWS Systems Manager, wo sie Verwaltungsaufgaben ausführen, Jobs ausführen, Daten verwalten und andere Shell-basierte Aktivitäten ausführen können. 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 (z. B.SSH), um auf Clusterressourcen zuzugreifen, Jobs auszuführen, Daten zu verwalten und andere Shell-basierte Aktivitäten durchzuführen.

Einrichtung für den AWS Parallel Computing Service

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

Themen

Melden Sie sich an für ein AWS-Konto

Wenn Sie noch keine haben AWS-Konto, führen Sie die folgenden Schritte aus, um eine zu erstellen.

Um sich für eine anzumelden AWS-Konto

1. Öffnen Sie https://portal.aws.amazon.com/billing/die Anmeldung.
2. Folgen Sie den Online-Anweisungen.

   Bei der Anmeldung müssen Sie auch einen Telefonanruf entgegennehmen und einen Verifizierungscode über die Telefontasten eingeben.

   Wenn Sie sich für eine anmelden AWS-Konto, Root-Benutzer des AWS-Kontoswird eine erstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Als bewährte Sicherheitsmethode weisen Sie einem Administratorbenutzer Administratorzugriff zu und verwenden Sie nur den Root-Benutzer, um Aufgaben auszuführen, die Root-Benutzerzugriff erfordern.

AWS sendet Ihnen nach Abschluss des Anmeldevorgangs eine Bestätigungs-E-Mail. Du kannst jederzeit deine aktuellen Kontoaktivitäten einsehen und dein Konto verwalten, indem du zu https://aws.amazon.com/gehst und Mein Konto auswählst.

Erstellen eines Benutzers mit Administratorzugriff

Nachdem Sie sich für einen angemeldet haben AWS-Konto, sichern Sie Ihren Root-Benutzer des AWS-Kontos AWS IAM Identity Center, aktivieren und erstellen Sie einen Administratorbenutzer, sodass Sie den Root-Benutzer nicht für alltägliche Aufgaben verwenden.

Sichern Sie Ihre Root-Benutzer des AWS-Kontos

1. Melden Sie sich AWS Management Consoleals Kontoinhaber an, indem Sie Root-Benutzer auswählen und Ihre AWS-Konto E-Mail-Adresse eingeben. Geben Sie auf der nächsten Seite Ihr Passwort ein.

   Hilfe bei der Anmeldung mit dem Root-Benutzer finden Sie unter Anmelden als Root-Benutzer im AWS-Anmeldung Benutzerhandbuch zu.
2. Aktivieren Sie die Multi-Faktor-Authentifizierung (MFA) für Ihren Root-Benutzer.

   Anweisungen finden Sie im Benutzerhandbuch unter Aktivieren eines virtuellen MFA Geräts für Ihren AWS-Konto IAM Root-Benutzer (Konsole).

Erstellen eines Benutzers mit Administratorzugriff

1. Aktivieren Sie IAM Identity Center.

   Anweisungen finden Sie unter Aktivieren AWS IAM Identity Center im AWS IAM Identity Center Benutzerhandbuch.
2. Gewähren Sie einem Benutzer in IAM Identity Center Administratorzugriff.

   Ein Tutorial zur Verwendung von IAM-Identity-Center-Verzeichnis als Identitätsquelle finden Sie unter Benutzerzugriff mit der Standardeinstellung konfigurieren IAM-Identity-Center-Verzeichnis im AWS IAM Identity Center Benutzerhandbuch.

Anmelden als Administratorbenutzer

- Um sich mit Ihrem IAM Identity Center-Benutzer anzumelden, verwenden Sie die Anmeldung, URL die an Ihre E-Mail-Adresse gesendet wurde, als Sie den IAM Identity Center-Benutzer erstellt haben.

Hilfe bei der Anmeldung mit einem IAM Identity Center-Benutzer finden Sie im AWS-Anmeldung Benutzerhandbuch unter Anmeldung beim AWS Zugangsportal.

Weiteren Benutzern Zugriff zuweisen

1. Erstellen Sie in IAM Identity Center einen Berechtigungssatz, der der bewährten Methode zur Anwendung von Berechtigungen mit den geringsten Rechten folgt.

   Anweisungen hierzu finden Sie unter Berechtigungssatz erstellen im AWS IAM Identity Center Benutzerhandbuch.
2. Weisen Sie Benutzer einer Gruppe zu und weisen Sie der Gruppe dann Single Sign-On-Zugriff zu.

   Eine genaue Anleitung finden Sie unter Gruppen hinzufügen im AWS IAM Identity Center Benutzerhandbuch.

Installieren Sie das AWS CLI

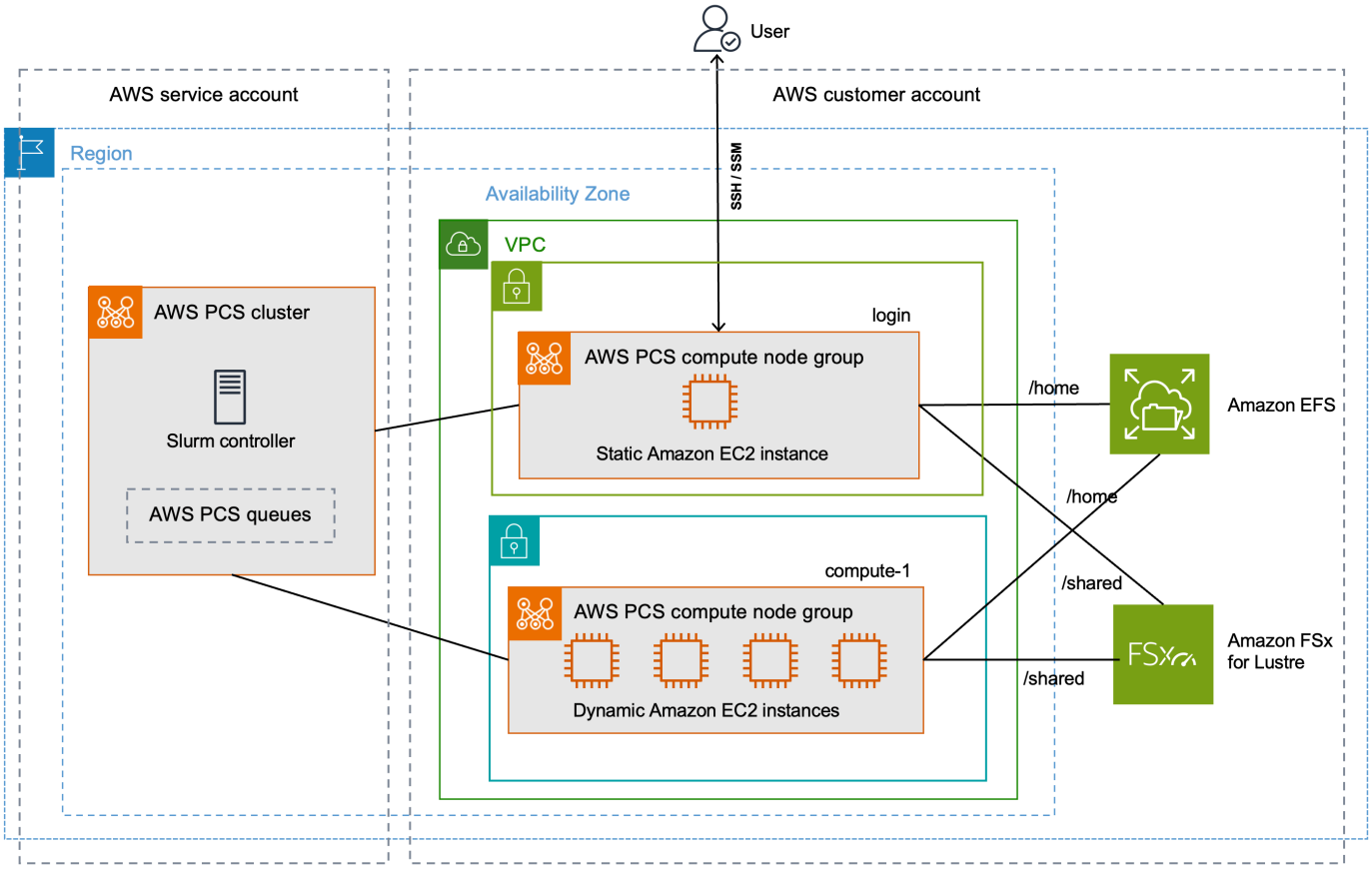
Sie müssen die neueste Version von verwenden AWS CLI. Weitere Informationen finden Sie unter Installation oder Aktualisierung auf die neueste Version von AWS CLI im AWS Command Line Interface Benutzerhandbuch für Version 2.

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

```
aws pcs help
```

Erste Schritte mit AWS PCS

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



Das Cluster-Design des Tutorials umfasst die folgenden Hauptkomponenten:

- A VPC und Subnetze, die die AWS PCSNetzwerkanforderungen erfüllen.
- Ein EFS Amazon-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 Compute-Knotengruppen.
  - 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.

Themen

Voraussetzungen für den Einstieg mit AWS PCS

Bevor Sie mit diesem Tutorial beginnen, installieren und konfigurieren Sie die folgenden Tools und Ressourcen, die Sie zum Erstellen und Verwalten eines AWS PCS Clusters benötigen.

- AWS CLI— Ein Befehlszeilentool für die Arbeit mit AWS Diensten, einschließlich AWS PCS. Weitere Informationen finden Sie unter Installation oder Aktualisierung auf die neueste Version von AWS CLI im AWS Command Line Interface Benutzerhandbuch für Version 2. Nach der Installation empfehlen wir AWS CLI, dass Sie es auch konfigurieren. Weitere Informationen finden Sie unter Configure the AWS CLI im AWS Command Line Interface Benutzerhandbuch für Version 2.
- Erforderliche IAM Berechtigungen — Der IAM Sicherheitsprinzipal, den Sie verwenden, muss über Berechtigungen für die Arbeit mit AWS PCS IAM Rollen, serviceverknüpften Rollen AWS CloudFormation VPC, a und verwandten Ressourcen verfügen. Weitere Informationen finden Sie Identity and Access Management für AWS Parallel Computing Service unter und Erstellen einer

dienstbezogenen Rolle im AWS Identity and Access Management Benutzerhandbuch. 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
```

- 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.

Erstellen Sie ein VPC UND-Subnetze für AWS PCS

Sie können ein VPC UND-Subnetze mit einer CloudFormation Vorlage erstellen. Gehen Sie wie folgt vor, URL um die CloudFormation Vorlage herunterzuladen, und laden Sie sie dann in die AWS CloudFormation Konsole hoch, um einen neuen CloudFormation Stack zu erstellen. Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch unter Verwenden der AWS 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 AWS CloudFormation Konsole die folgenden Optionen ein. Sie können die in der Vorlage bereitgestellten Standardwerte verwenden.

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

    ```
    hpc-networking
    ```
- Unter Parameter:
  - Unter VPC:
    - Geben Sie CidrBlockunter Folgendes 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 aus
  - 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 AWS CloudFormation dadurch IAM Ressourcen erstellt werden könnten.

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

Suchen Sie die Standardsicherheitsgruppe für den Cluster VPC

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

- Navigieren Sie zur VPCAmazon-Konsole.
- Wählen Sie im VPCDashboard die Option Filtern nach ausVPC.
  - Wählen Sie die VPC Stelle aus, mit der der Name beginnt`hpc-networking`.
  - 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 erstellen für AWS PCS

AWS PCSstützt 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 für zwei Sicherheitsgruppen.

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

Erstellen Sie die Sicherheitsgruppen für AWS PCS

Mit dieser CloudFormation Vorlage können Sie ein VPC und Subnetze erstellen. Gehen Sie wie folgt vor, URL um die CloudFormation Vorlage herunterzuladen, und laden Sie sie dann in die AWS CloudFormation Konsole hoch, um einen neuen CloudFormation Stack zu erstellen. Weitere Informationen finden Sie im AWS CloudFormation Benutzerhandbuch unter Verwenden der AWS 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 unter die VPC Stelle aus VpcId, mit der der Name beginnt`hpc-networking`.
  - (Optional) Geben Sie unter ClientIpCidreinen restriktiveren IP-Bereich für die SSH Sicherheitsgruppe für eingehende Nachrichten 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 PUBLIC x.x.x.x durch Ihre eigene 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 Zugriff SSH 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 (unterstützt AWS PCS derzeit Slurm) in einem Subnetz eines neuen oder bestehenden. VPC Der Cluster akzeptiert und plant Jobs und startet auch die Rechenknoten (EC2Instances), die diese Jobs verarbeiten.

Um Ihren Cluster zu erstellen

1. Öffnen Sie die AWS PCSKonsole und wählen Sie Create Cluster aus.

2. Geben Sie im Abschnitt Cluster-Setup die folgenden Felder ein:
   - Clustername — Geben Sie ein `get-started`
   - Controller-Größe — Wählen Sie Klein
3. Wählen Sie im Bereich Netzwerk Werte für die folgenden Felder aus:
   - VPC— Wählen Sie den VPC Benannten `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 aus `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.

Gemeinsamer Speicher für AWS PCS in Amazon Elastic File System erstellen

Amazon Elastic File System (AmazonEFS) 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 Home-Verzeichnis zwischen den Cluster-Knoten bereitzustellen. Erstellen Sie ein EFS Dateisystem im gleichen Format VPC wie Ihr Cluster.

Um Ihr EFS Amazon-Dateisystem zu erstellen

1. Gehen Sie zur EFSAmazon-Konsole.
2. Stellen Sie sicher, dass es auf die gleiche Einstellung eingestellt ist AWS-Region , auf der Sie es versuchen werden AWS PCS.

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 benannte `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.

Gemeinsamer Speicher für AWS PCS in Amazon FSx for Lustre erstellen

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, Hochleistungsrechnen (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 demselben Format VPC wie Ihr Cluster.

Um Ihr FSx for Lustre-Dateisystem zu erstellen

1. Gehen Sie zur FSxAmazon-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 Option 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 benannte `hpc-networking:Large-Scale-HPC`
  - Behalten Sie für VPCSicherheitsgruppen den Namen der Sicherheitsgruppe bei `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 (EC2Instanzen), die AWS PCS gestartet und verwaltet werden. Wenn Sie eine Rechenknotengruppe definieren, geben Sie allgemeine Merkmale wie EC2 Instanztypen, minimale und maximale Anzahl von Instanzen, VPC Zielsubnetze, bevorzugte Kaufoption und benutzerdefinierte Startkonfiguration an. AWS PCSStartet, verwaltet und beendet Rechenknoten in einer Rechenknotengruppe effizient gemäß

diesen Einstellungen. 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 verwenden, AWS Management Console um eine Rolle für Amazon zu erstellenEC2, erstellt die Konsole automatisch ein Instance-Profil und weist diesem den gleichen Namen wie die Rolle zu. Weitere Informationen finden Sie unter Verwenden von Instance-Profilen im AWS Identity and Access Management Benutzerhandbuch.

Im folgenden Verfahren verwenden Sie die, AWS Management Console um eine Rolle für Amazon zu erstellenEC2, die auch das Instance-Profil für Ihre Compute-Knotengruppen erstellt.

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 Create Policy (Richtlinie erstellen) aus.
  - Wählen Sie unter Berechtigungen angeben für den Policy-Editor die Option JSON.
  - Ersetzen Sie den Inhalt des Texteditors durch Folgendes:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "pcs:RegisterComputeNodeGroupInstance"
            ],
```

```
            "Resource": "*",
            "Effect": "Allow"
        }
    ]
}
```

- Wählen Sie Weiter.
- Geben Sie unter Überprüfen und erstellen als Richtlinienname den Wert ein`AWSPCS-getstarted-policy`.
- Wählen Sie Create Policy (Richtlinie erstellen) aus.

- Wählen Sie unter Access management (Zugriffsverwaltung) Roles (Rollen) aus.
- Wählen Sie Rolle erstellen.
- Unter Vertrauenswürdige Entität auswählen:
  - Wählen Sie für Vertrauenswürdigen Entitätstyp die Option AWS Dienst aus
  - Wählen Sie unter Anwendungsfall die Option aus EC2.
    - Wählen Sie dann unter Wählen Sie einen Anwendungsfall für den angegebenen Dienst die Option aus EC2.
  - Wählen Sie Weiter.
- 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 Berechtigungsrichtlinien nach A. mazonSSMManaged InstanceCore
  - Markieren Sie das Kästchen neben A mazonSSMManaged InstanceCore, um es der Rolle hinzuzufügen.
  - Wählen Sie Weiter.
- Unter Name überprüfen und erstellen Sie:
  - Unter Rollendetails:
    - Geben Sie für Role name (Rollenname) den Namen `AWSPCS-getstarted-role` ein.
  - Wählen Sie Create role (Rolle erstellen) aus.

Startvorlagen erstellen für AWS PCS

Wenn Sie eine Compute-Knotengruppe erstellen, stellen Sie eine EC2 Startvorlage bereit, die zur Konfiguration der von ihr gestarteten EC2 Instances AWS PCS verwendet wird. 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 zum Erstellen von Anmeldeknoten und die andere zum Erstellen 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

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

- 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 Ihrem Cluster benannt istVPC.
    - 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 key pair aus.

- Unter Dateisysteme
  - Geben Sie für EfsFilesystemIddie Dateisystem-ID des EFS Dateisystems 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 AWS 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 (EC2Instanzen), die AWS PCS gestartet und verwaltet werden. Wenn Sie eine Rechenknotengruppe definieren, geben Sie allgemeine Merkmale wie EC2 Instanztypen, minimale und maximale Anzahl von Instanzen, VPC Zielsubnetze, bevorzugte Kaufoption und benutzerdefinierte Startkonfiguration an. AWS PCSStartet, verwaltet und beendet Rechenknoten in einer Rechenknotengruppe effizient gemäß diesen Einstellungen.

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 PCSKonsole und navigieren Sie zu Clusters.
- Wählen Sie den Cluster mit dem Namen aus `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:
  - EC2Startvorlage — Wählen Sie die Startvorlage aus, deren Name steht `login-getstarted-lt`
  - IAMInstanzprofil — Wählen Sie das angegebene Instanzprofil `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 Mindestanzahl der Instanzen ein. Geben 1 Sie für Max. Anzahl der Instanzen den Wert ein.
- Geben Sie unter Zusätzliche Einstellungen Folgendes an:
  - AMIID — Wählen Sie die AMI Stelle aus, mit der der Name beginnt `aws-pcs-sample_ami-amzn2-x86_64-slurm-23.11`
- Wählen Sie Compute-Knotengruppe erstellen 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 PCSKonsole 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:

- EC2Startvorlage — Wählen Sie die Startvorlage aus, deren Name steht `compute-getstarted-lt`
- IAMInstanzprofil — Wählen Sie das angegebene Instanzprofil `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:
  - AMIID — Wählen Sie die AMI Stelle aus, mit der der Name beginnt`aws-pcs-sample_ami-amzn2-x86_64-slurm-23.11`.
- Wählen Sie Compute-Knotengruppe erstellen 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 er für die Ausführung auf einer Rechenknotengruppe geplant ist. 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 PCSKonsole.

- 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 Anmeldeknoten herzustellen

- Öffnen Sie die AWS PCSKonsole 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 EC2Amazon-Konsole.
  - Wählen Sie Instances.
  - Suchen Sie nach EC2 Instances mit dem folgenden Tag. Ersetzen *node-group-id* mit dem 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 verwendenSSH.

    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 SSHClient aus.
    - Folgen Sie den Anweisungen auf 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 2. Sie müssen dies nicht tun, wenn Sie über eine Verbindung hergestellt habenSSH.

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

Arbeiten Sie mit gemeinsam genutzten Dateisystemen

Sie können mit dem Befehl überprüfen, ob EFS das Dateisystem und die Dateisysteme FSx für 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

Themen

- 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 aus in AWS PCS

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 *job-id* im folgenden Befehl mit der 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 aus AWS PCS

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

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 MPI Open-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 PCSKonsole.
- 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 IAMKonsole.
  - Wählen Sie Roles.
  - Wählen Sie die Rolle mit dem Namen AWSPCS-getstarted-role und anschließend Löschen aus.
  - 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 AWS 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 EFSAmazon-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 FSxAmazon-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 AWS CloudFormation -Konsole.
  - Wählen Sie den Stack mit dem Namen getstarted-sg aus.
  - Wählen Sie Löschen.
- Öffnen Sie die AWS CloudFormation -Konsole.
  - Wählen Sie den Stack mit dem Namen hpc-networking aus.
  - Wählen Sie Löschen.

Arbeiten mit AWS PCS

Dieses Kapitel enthält Informationen und Anleitungen zur Verwendung AWS PCS.

Themen

AWS PCSCluster

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 in den HPC Systemplaner integriert sind, um EC2 Amazon-Instances bereitzustellen und zu verwalten.
- Komponenten, die in den HPC Systemplaner integriert sind, um Protokolle und Metriken an Amazon CloudWatch zu übertragen.

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 PCSstellt elastische Netzwerkschnittstellen in Ihrem VPC Amazon-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 im AWS Parallel Computing Service erstellen

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

Voraussetzungen

- Ein vorhandenes Subnetz VPC und ein Subnetz, das die Anforderungen erfülltAWS PCSNetzwerkbetrieb. Bevor Sie einen Cluster für Produktionszwecke bereitstellen, sollten Sie sich mit den Anforderungen VPC und den Subnetzanforderungen gründlich vertraut machen. Informationen zum Erstellen eines VPC UND-Subnetzes finden Sie unter. Erstellen eines VPC für Ihren AWS PCS Cluster
- Ein IAMPrincipal mit Berechtigungen zum Erstellen und Verwalten von AWS PCS Ressourcen. Weitere Informationen finden Sie unter Identity and Access Management für AWS Parallel Computing Service.

Erstellen Sie einen AWS PCS Cluster

Sie können das AWS Management Console oder verwenden AWS CLI , um einen Cluster zu erstellen.

AWS Management Console

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. AWS PCSunterstützt derzeit Slurm 23.11. Weitere Informationen finden Sie unter Slurm-Versionen in AWS PCS.
- 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 unterAuswahl einer AWS PCS Clustergröße.

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

   - VPC— Wählen Sie ein vorhandenesVPC, das den AWS PCS Anforderungen entspricht. Weitere Informationen finden Sie unter AWS PCSVPCund Subnetzanforderungen und Überlegungen. Nachdem Sie den Cluster erstellt haben, können Sie ihn nicht mehr ändernVPC. Wenn keine aufgeführt VPCs sind, müssen Sie zuerst einen erstellen.
   - Subnetz — Alle verfügbaren Subnetze in den ausgewählten Subnetzen VPC werden aufgelistet. Wählen Sie zwei in verschiedenen Availability Zones aus. Jedes Subnetz muss die AWS PCS Subnetzanforderungen erfüllen. Weitere Informationen finden Sie unter AWS PCSVPCund Subnetzanforderungen und Überlegungen. 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 Sie den Netzwerkschnittstellen zuordnen AWS PCS möchten, die für Ihren Cluster erstellt werden. 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.

4. (Optional) Unter Verschlüsselung können Sie einen benutzerdefinierten Schlüssel zur Verschlüsselung Ihrer Controller-Daten definieren, indem Sie die folgenden Felder festlegen:

   - KMSSchlüssel-ID — Geben Sie an, ob Sie `aws/pcs` den KMS Schlüssel verwenden möchten, der PCS erstellt. Wählen Sie einen vorhandenen KMS Schlüsselalias aus, um einen benutzerdefinierten KMS Schlüssel zu verwenden. Beachten Sie, dass das

Konto, mit dem der Cluster erstellt wurde, über `kms:Decrypt` Berechtigungen für den benutzerdefinierten KMS Schlüssel verfügen muss.

5. (Optional) Im Abschnitt Slurm-Konfiguration können Sie Slurm-Konfigurationsoptionen angeben, die die Standardeinstellungen außer Kraft setzen, die wie folgt festgelegt wurden: AWS PCS

   - Leerlaufzeit herunterskalieren — Damit wird gesteuert, wie lange dynamisch bereitgestellte Rechenknoten aktiv bleiben, nachdem die ihnen zugewiesenen Jobs abgeschlossen oder beendet wurden. Wenn Sie diesen Wert auf einen längeren Wert setzen, ist es wahrscheinlicher, dass ein nachfolgender Job auf dem Knoten ausgeführt werden kann, was jedoch zu höheren Kosten führen kann. Ein kürzerer Wert senkt die Kosten, kann jedoch den Anteil der Zeit erhöhen, die Ihr HPC System mit der Bereitstellung von Knoten verbringt, anstatt Aufträge auf ihnen auszuführen.
   - Prolog — Dies ist ein vollständig qualifizierter Pfad zu einem Prolog-Skriptverzeichnis auf Ihren Compute-Knotengruppen-Instances. Dies entspricht der Prolog-Einstellung in Slurm. Beachten Sie, dass dies ein Verzeichnis sein muss, kein Pfad zu einer bestimmten ausführbaren Datei.
   - Epilog — Dies ist ein vollständig qualifizierter Pfad zu einem Epilog-Skriptverzeichnis auf Ihren Compute-Knotengruppen-Instances. Dies entspricht der Epilog-Einstellung in Slurm. Beachten Sie, dass dies ein Verzeichnis sein muss, kein Pfad zu einer bestimmten ausführbaren Datei.
   - Typparameter auswählen — Dies hilft bei der Steuerung des von Slurm verwendeten Algorithmus zur Ressourcenauswahl. Wenn Sie diesen Wert auf setzen, `CR_CPU_Memory` wird die speicherorientierte Planung aktiviert, wenn Sie ihn auf setzen, `CR_CPU` wird die reine Planung aktiviertCPU. Dieser Parameter entspricht der SelectTypeParametersEinstellung in Slurm, wo der Wert auf by gesetzt `SelectType` ist. `select/cons_tres` AWS PCS

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 Cluster AWS PCS erstellt wird. Dieser Vorgang kann einige Minuten dauern.

> **Important**
>
> AWS-Region Pro `Creating` Bundesstaat kann es nur einen Cluster geben AWS-Konto. AWS PCSgibt 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:

   - Ersetzen *`region`* mit der ID des Clusters AWS-Region , in dem Sie Ihren Cluster erstellen möchten, z. `us-east-1` B.
   - Ersetzen *`my-cluster`* mit einem Namen für Ihren Cluster. Der Name darf nur alphanumerische Zeichen (wobei die Groß- und Kleinschreibung beachtet werden muss) und Bindestriche enthalten. Es 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.
   - Ersetzen *`23.11`* mit jeder unterstützten Version von Slurm.

     > **Note**
     >
     > AWS PCSunterstützt derzeit Slurm 23.11.

   - Ersetzen *`SMALL`* mit jeder unterstützten 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 unterAuswahl einer AWS PCS Clustergröße.
   - 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 die an`securityGroupIds`, die Sie den Netzwerkschnittstellen zuordnen AWS PCS möchten, die es für Ihren Cluster erstellt. Die Sicherheitsgruppen müssen sich im selben VPC 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.

- Optional können Sie das Verhalten von Slurm feinabstimmen, indem Sie eine `--slurm-configration` Option hinzufügen. Mit können Sie beispielsweise die Leerlaufzeit beim Herunterfahren auf 60 Minuten (3600 Sekunden) festlegen. `--slurm configuration scaleDownIdeTime=3600`
- Optional können Sie einen benutzerdefinierten KMS Schlüssel angeben, mit dem Sie die Daten Ihres Controllers verschlüsseln können. `--kms-key-id` *kms-key* *kms-key*Ersetzen Sie ihn durch eine vorhandene KMS ARN Schlüssel-ID oder einen Alias. Beachten Sie, dass das Konto, mit dem der Cluster erstellt wurde, über `kms:Decrypt` Berechtigungen für den benutzerdefinierten KMS Schlüssel verfügen muss.

```
aws pcs create-cluster --region region \
    --cluster-name my-cluster \
    --scheduler type=SLURM,version=23.11 \
    --size SMALL \
    --networking subnetIds=subnet-ExampleId1,securityGroupIds=sg-ExampleId1
```

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 PCSgibt 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.

Löschen eines Clusters in AWS PCS

Dieses Thema bietet einen Überblick über das Löschen eines AWS PCS Clusters.

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 Management Console oder verwenden AWS CLI , um einen Cluster zu löschen.

AWS Management Console

Löschen eines Clusters

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

AWS CLI

Löschen eines Clusters

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

- Ersetzen *my-cluster* mit dem Namen oder der 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
```

Auswahl einer AWS PCS Clustergröße

AWS PCSbietet 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 sollen
- Die Anzahl der aktiven Jobs und Warteschlangenjobs, die Sie voraussichtlich auf dem Cluster ausführen werden

Größe des Slurm-Clusters	Anzahl der verwalteten Instanzen	Anzahl der aktiven Jobs und Jobs in der Warteschlange
Small	Bis zu 32	Bis zu 256
Mittelschwer	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

AWS PCSErstellt im Rahmen der Clustererstellung 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 PCSkonfiguriert 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 PCSspeichert das Clustergeheimnis als verwaltetes Geheimnis mit dem Präfix `pcs!` in AWS Secrets Manager. Die Kosten für das Secret sind in der Nutzungsgebühr enthalten AWS PCS.

> **Warning**
>
> Ändern Sie Ihr Clustergeheimnis nicht. AWS PCSkann nicht mit Ihrem Cluster kommunizieren, wenn Sie Ihr Clustergeheimnis ändern. AWS PCSunterstützt die Rotation des Clustergeheimnisses nicht. Sie müssen einen neuen Cluster erstellen, wenn Sie Ihr Clustergeheimnis ändern müssen.

Inhalt

Finden Sie das Geheimnis des Slurm-Clusters

Sie können AWS PCS verwaltete Geheimnisse über die AWS Secrets Manager Konsole oder direkt von oder API mithilfe von AWS PCS Tags finden.

Wird verwendet AWS Secrets Manager , um das Clustergeheimnis zu finden

AWS Management Console

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 AWS PCS Clustergeheimnis ist auch 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
```

Wird verwendet AWS PCS, um das Cluster-Geheimnis zu finden

Sie können das verwenden AWS CLI , um nach einem AWS PCS Clustergeheimnis zu suchen. ARN 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": "pcsdemo",
        "id": "s3431v9rx2",
        "arn": "arn:aws:pcs:us-east-1:012345678901:cluster/s3431v9rx2",
        "status": "ACTIVE",
        "createdAt": "2024-07-12T15:32:27.225136+00:00",
        "modifiedAt": "2024-07-12T15:32:27.225136+00:00",
        "scheduler": {
            "type": "SLURM",
            "version": "23.11"
        },
        "size": "SMALL",
        "networking": {
            "subnetIds": [
                "subnet-0123456789abcdef"
            ],
            "securityGroupIds": [
                "sg-0123456789abcde"
            ]
        },
        "endpoints": [
            {
                "type": "SLURMCTLD",
                "privateIpAddress": "127.0.0.1",
                "port": "6817"
            }
        ],
        "secretArn": "arn:aws:secretsmanager:us-east-1:012345678901:secret:pcs!slurm-
secret-s3431v9rx2-FN7tJF",
        "secretVersion": "ff58d1fd-070e-4bbc-98a0-64ef967cebcc"
    }
```

```
}
```

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 unterStandalone-Instanzen als AWS PCS Login-Knoten verwenden.

Berechtigungen

Sie verwenden einen IAM Principal, um das Geheimnis des Slurm-Clusters abzurufen. Der IAM Principal muss die Erlaubnis haben, das Geheimnis zu lesen. Weitere Informationen finden Sie im AWS Identity and Access Management Benutzerhandbuch unter Begriffe und Konzepte für Rollen.

Die folgende IAM Beispielrichtlinie 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"
        }
    ]
}
```

AWS PCSKnotengruppen berechnen

Eine AWS PCS Rechenknotengruppe ist eine logische Sammlung von Knoten (EC2Amazon-Instances). Diese Knoten können zur Ausführung von Rechenjobs sowie zum Bereitstellen eines interaktiven, Shell-basierten Zugriffs auf ein HPC System verwendet werden. Eine Compute-Knotengruppe besteht aus Regeln für die Erstellung von Knoten, einschließlich der zu verwendenden EC2 Amazon-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, AWS PCS werden die der Rechenknotengruppe zugewiesenen Ressourcen entsprechend aktualisiert.

Themen

- Erstellen einer Compute-Knotengruppe in AWS PCS
- Aktualisierung einer AWS PCS Compute-Knotengruppe
- Löschen einer Compute-Knotengruppe in AWS PCS
- Suchen nach Instanzen der Compute-Knotengruppe in AWS PCS

Erstellen 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 Parallel Computing Service (AWS PCS) erstellen. Wenn Sie zum ersten Mal eine Rechenknotengruppe in erstellen AWS PCS, empfehlen wir Ihnen, das Tutorial unter zu befolgenErste Schritte mit AWS PCS. Das Tutorial kann Ihnen helfen, ein funktionierendes HPC System zu erstellen, ohne auf alle verfügbaren Optionen und Systemarchitekturen eingehen zu müssen, die möglich sind.

Voraussetzungen

- Ausreichende Servicekontingente, um die gewünschte Anzahl von EC2 Instanzen in Ihrem zu AWS-Region starten. Sie können den verwenden AWS Management Console, um eine Erhöhung Ihrer Servicekontingenten zu überprüfen und zu beantragen.
- Ein vorhandenes VPC und ein oder mehrere Subnetze, die die AWS PCS Netzwerkanforderungen erfüllen. Wir empfehlen, dass Sie sich gründlich mit diesen Anforderungen vertraut machen, bevor Sie einen Cluster für den Produktionseinsatz bereitstellen. Weitere Informationen finden Sie unter AWS PCSVPCund Subnetzanforderungen und Überlegungen. Sie können auch eine CloudFormation Vorlage verwenden, um ein VPC End-Subnetz zu erstellen. AWS bietet ein HPC Rezept für die CloudFormation Vorlage. Weitere Informationen finden Sie aws-hpc-recipesunter GitHub.
- Ein IAM Instanzprofil mit Berechtigungen zum Aufrufen der AWS PCS `RegisterComputeNodeGroupInstance` API Aktion und zum Zugriff auf alle anderen AWS Ressourcen, die für Ihre Knotengruppen-Instances erforderlich sind. Weitere Informationen finden Sie unter IAMInstanzprofile für AWS Parallel Computing Service.
- Eine Startvorlage für Ihre Knotengruppen-Instances. Weitere Informationen finden Sie unter Verwenden von EC2 Amazon-Startvorlagen mit AWS PCS.
- Um eine Compute-Knotengruppe zu erstellen, die Amazon EC2 Spot-Instances verwendet, müssen Sie die AWSServiceRoleForEC2Spotserviceverknüpfte Rolle in Ihrer AWS-Konto haben. Weitere Informationen finden Sie unter Amazon EC2 Spot-Rolle für AWS PCS.

Erstellen Sie eine Rechenknotengruppe in AWS PCS

Sie können eine Rechenknotengruppe mit dem AWS Management Console oder dem erstellen AWS CLI.

AWS Management Console

Um Ihre Compute-Knotengruppe mithilfe der Konsole zu erstellen

1. Öffnen Sie die AWS PCSKonsole.
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. EC2Startvorlage — 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 EC2 Amazon-Startvorlagen mit AWS PCS Sie eine erstellen.

   > **Important**
   >
   > AWS PCSerstellt 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. EC2Version der Startvorlage — Wählen Sie eine Version Ihrer benutzerdefinierten Startvorlage aus. Sie können eine bestimmte Version wählen, wodurch die Reproduzierbarkeit verbessert werden kann. 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 einer AWS PCS Compute-Knotengruppe.

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

   d. IAMInstanzprofil — Wählen Sie ein Instanzprofil für die Knotengruppe aus. Ein Instanzprofil gewährt der Instanz Berechtigungen für den sicheren Zugriff auf AWS Ressourcen und Dienste. Falls Sie noch kein Konto vorbereitet haben, erfahren IAMInstanzprofile für AWS Parallel Computing Service Sie unter, wie Sie eines erstellen.

e. Subnetze — Wählen Sie ein oder mehrere Subnetze in dem Bereich aus, in VPC dem 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 den Subnetzen 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 (x864_64 oder arm64) und dieselbe Anzahl von haben. vCPUs Wenn dies bei den Instanzen der Fall istGPUs, müssen alle Instanztypen dieselbe Anzahl von haben. GPUs

g. Skalierungskonfiguration — Geben Sie die Mindest- und Höchstanzahl von Instanzen für die Knotengruppe an. Sie können entweder eine statische Konfiguration definieren, bei der eine feste Anzahl von Knoten ausgeführt wird, oder eine dynamische Konfiguration, bei der bis zu die maximale Anzahl von Knoten ausgeführt werden kann. Bei einer statischen Konfiguration legen Sie für Minimum und Maximum dieselbe Zahl fest, die größer als Null ist. Legen Sie für eine dynamische Konfiguration die Mindestanzahl der Instanzen auf Null und die maximale Anzahl der Instanzen auf eine Zahl größer als Null fest. AWS PCSunterstützt keine Rechenknotengruppen mit einer Mischung aus statischen und dynamischen Instanzen.

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

   a. Kaufoption — Wählen Sie zwischen Spot- und On-Demand-Instances.

   b. Zuweisungsstrategie — Wenn Sie die Spot-Kaufoption ausgewählt haben, können Sie angeben, wie Spot-Kapazitätspools beim Start 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 Option On-Demand-Kauf ausgewählt haben.

6. (Optional) Geben Sie im Abschnitt „SlurmBenutzerdefinierte Einstellungen“ die folgenden Werte an:

   a. Gewicht — Dieser Wert legt die Priorität der Knoten in der Gruppe für Planungszwecke fest. Knoten mit niedrigerer Gewichtung haben eine höhere Priorität, und die Einheiten sind willkürlich. Weitere Informationen finden Sie in der Slurm Dokumentation unter Gewicht.

b. Realer Speicher — Dieser Wert legt die Größe (in GB) des realen Speichers auf Knoten in der Knotengruppe fest. Er ist für die Verwendung in Verbindung mit der `CR_CPU_Memory` Option in der Slurm Cluster-Konfiguration unter vorgesehen AWS PCS. Weitere Informationen finden Sie RealMemoryin der Slurm Dokumentation.

7. (Optional) Fügen Sie unter Tags beliebige Tags zu Ihrer Compute-Knotengruppe hinzu.
8. Wählen Sie Compute-Knotengruppe erstellen aus. Im Feld Status wird angezeigt, `Creating` während AWS PCS die Knotengruppe bereitgestellt wird. Dies kann mehrere Minuten dauern.

Als nächster Schritt wird empfohlen

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

> **Important**
>
> AWS PCSerstellt eine verwaltete Startvorlage für jede Rechenknotengruppe. 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. Ersetzen *launch-template-version* mit einer bestimmten Version der Startvorlage, wenn Sie Ihre Knotengruppe einer bestimmten Version zuordnen möchten.
7. Ersetzen *arn:InstanceProfile*mit dem ARN Ihres IAM Instanzprofils. Falls Sie noch keines vorbereitet haben, finden Sie Verwenden von EC2 Amazon-Startvorlagen mit AWS PCS weitere Informationen unter.
8. Ersetzen *min-instances* and *max-instances* mit ganzzahligen Werten. Sie können entweder eine statische Konfiguration definieren, bei der eine feste Anzahl von Knoten ausgeführt wird, oder eine dynamische Konfiguration, bei der bis zu die maximale Anzahl von Knoten ausgeführt werden kann. Bei einer statischen Konfiguration legen Sie für Minimum und Maximum dieselbe Zahl fest, die größer als Null ist. Legen Sie für eine dynamische Konfiguration die Mindestanzahl der Instanzen auf Null und die maximale Anzahl der Instanzen auf eine Zahl größer als Null fest. AWS PCSunterstützt keine Rechenknotengruppen mit einer Mischung aus statischen und dynamischen Instanzen.
9. Ersetzen *t3.large* mit einem anderen Instanztyp. Sie können weitere Instanztypen hinzufügen, indem Sie eine Liste mit `instanceType` Einstellungen angeben. Zum Beispiel *--instance-configs instanceType=c6i.16xlarge,instanceType=c6a.16xlarge*. Alle Instance-Typen müssen dieselbe Prozessorarchitektur (x864_64 oder arm64) und dieselbe Anzahl von haben. vCPUs Wenn dies bei den Instanzen der Fall istGPUs, müssen alle Instanztypen dieselbe Anzahl von haben. GPUs

```
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
```

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 einen AMI enthält oder ob Sie diesen Wert überschreiben möchten. Beachten Sie, dass die für den Knoten AMI verwendete Gruppe kompatibel sein muss mit AWS PCS. Sie können auch ein von AMI bereitgestelltes Beispiel auswählen AWS. Weitere Informationen zu diesem Thema finden Sie unterAmazon Machine Images (AMIs) für AWS PCS.
- Mithilfe von können Sie zwischen On-Demand-Instances (`ONDEMAND`) und Spot-Instances (`SPOT`) wählen`--purchase-option`. On-Demand ist die Standardeinstellung. Wenn Sie Spot-Instances wählen, können Sie `--allocation-strategy` damit auch definieren, wie Spot-Kapazitätspools AWS PCS ausgewählt werden, wenn Instances in der Knotengruppe gestartet werden. Weitere Informationen finden Sie unter Zuweisungsstrategien für Spot-Instances im Amazon Elastic Compute Cloud-Benutzerhandbuch.
- 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 AWS PCS in Ihrer Slurm Konfiguration verwendet werden. Weitere Informationen finden Sie RealMemoryin 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 einer AWS PCS Compute-Knotengruppe

Dieses Thema bietet einen Überblick über die verfügbaren Optionen und beschreibt, was bei der Aktualisierung einer AWS PCS Compute-Knotengruppe zu beachten ist.

Optionen für die Aktualisierung einer AWS PCS Rechenknotengruppe

Durch die Aktualisierung einer AWS PCS Compute-Knotengruppe können Sie die Eigenschaften von Instances ändern AWSPCS, die von gestartet wurden, sowie die Regeln dafür, wie diese Instances gestartet werden. Sie können AMI beispielsweise die vier Knotengruppen-Instances durch eine andere ersetzen, auf der andere Software installiert ist. Oder Sie können Sicherheitsgruppen aktualisieren, um die eingehende oder ausgehende Netzwerkkonnektivität zu ändern. Sie können auch die Skalierungskonfiguration ändern oder sogar die bevorzugte Kaufoption für Spot-Instances oder für Spot-Instances ändern.

Die folgenden Knotengruppeneinstellungen können nach der Erstellung nicht geändert werden:

- Name
- Instances

Überlegungen beim Aktualisieren einer AWS PCS Compute-Knotengruppe

Compute-Knotengruppen definieren EC2 Instanzen, 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. Beachten Sie Folgendes, wenn Sie Ihre Compute-Knotengruppe aktualisieren, um ihr Verhalten (oder das ihrer Knoten) zu ändern:

- 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 IAM Instanzprofil, das von einer Compute-Knotengruppe verwendet wird, Berechtigungen hinzufügen oder daraus entfernen, wird dies sofort auf laufende und future Instances wirksam.
- Um die von einer Compute-Knotengruppe AMI verwendeten Instanzen zu ändern, aktualisieren Sie die Compute-Knotengruppe (oder ihre Startvorlage) so, dass sie die neuen Instanzen verwendet, AMI und warten Sie AWS PCS, bis die Instanzen ersetzt werden.
- AWS PCSersetzt bestehende Instanzen in der Knotengruppe nach einem Aktualisierungsvorgang für die Knotengruppe. Wenn auf einem Knoten Jobs ausgeführt werden, können diese Jobs abgeschlossen werden, bevor der Knoten AWS PCS ersetzt wird. Interaktive Benutzerprozesse (z. B. auf Anmeldeknoteninstanzen) werden beendet. Der Status der Knotengruppe kehrt zu dem `Active` Zeitpunkt zurück, an dem die Instances als Ersatz AWS PCS markiert werden, 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, werden Knoten aus Slurm AWS PCS entfernt, um das neue Maximum zu erreichen. AWS PCSbeendet laufende Instanzen, die den entfernten Slurm-Knoten zugeordnet sind. Die laufenden Jobs auf den entfernten Knoten schlagen fehl und kehren in ihre Warteschlangen zurück.
- AWS PCSerstellt eine verwaltete Startvorlage für jede Rechenknotengruppe. 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 PCS.

Um eine AWS PCS Compute-Knotengruppe zu aktualisieren

Sie können eine Knotengruppe mithilfe der AWS Management Console oder der aktualisieren AWSCLI.

AWS Management Console

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 Compute-Knotengruppe 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.

5. Wählen Sie Aktualisieren. 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. Ersetzen *region-code* mit der AWS Region, in der Sie Ihren Cluster erstellen möchten.

   b. Ersetzen *my-node-group* mit dem Namen oder `computeNodeGroupId` für Ihre Compute-Knotengruppe.

   c. Ersetzen *my-cluster* mit dem 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
```

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 die AMI von Ihnen gewählte 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 Compute-Knotengruppe in löschen AWS PCS.

Überlegungen beim Löschen einer Compute-Knotengruppe

Compute-Knotengruppen definieren EC2 Instanzen, 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 Instanzen werden beendet. Dadurch werden Jobs storniert, die auf diesen Instanzen 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 Eine AWS PCS Warteschlange aktualisieren.

Löschen Sie die Compute-Knotengruppe

Sie können das AWS Management Console oder verwenden AWS CLI , um eine Rechenknotengruppe zu löschen.

AWS Management Console

Um eine Compute-Knotengruppe zu löschen

1. Öffnen Sie die AWS PCSKonsole.
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.
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 *`region-code`* mit dem, in dem sich AWS-Region Ihr Cluster befindet.
  - Ersetzen *`my-node-group`* mit dem Namen oder der ID Ihrer Compute-Knotengruppe.
  - Ersetzen *`my-cluster`* mit dem Namen oder der 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.

Suchen nach Instanzen der Compute-Knotengruppe in AWS PCS

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

AWS Management Console

Um Ihre Compute-Knotengruppen-Instanzen zu finden

1. Öffnen Sie die AWS PCSKonsole.
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 EC2Konsole und wählen Sie Instances aus.
6. Suchen Sie nach den Instances mit dem folgenden Tag. Ersetzen *`node-group-id`* mit der ID (nicht dem Namen) Ihrer Compute-Knotengruppe.

   ```
   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.
- `running`Ersetzen Sie durch andere Instanzstatus, z. B. `pending` oder`terminated`, um EC2 Instanzen in anderen Bundesstaaten 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 Instanz 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 API Cloud-Referenz.

Verwenden von EC2 Amazon-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 PCSbeinhaltet Startvorlagen

als flexible Methode zur Konfiguration von Rechenknotengruppen. Wenn Sie eine Knotengruppe erstellen, stellen Sie eine Startvorlage bereit. AWS PCSerstellt 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 erforderlich sind, können Sie eine Vorlage für die Verwendung mit AWS PCS erstellen. Weitere Informationen zu Startvorlagen finden Sie im EC2Amazon-Benutzerhandbuch unter Starten einer Instance von einer Startvorlage aus starten.

Themen

Übersicht

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 kompatibel mit AWS PCS, es gibt jedoch einige Ausnahmen.

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

- Instanztyp/Instanztypattribute angeben (`InstanceRequirements`) — unterstützt AWS PCS keine attributbasierte Instanzauswahl.
- Instanztyp (`InstanceType`) — Geben Sie Instanztypen an, wenn Sie eine Knotengruppe erstellen.
- Erweiterte IAM Details/Instanzprofil (`IamInstanceProfile`) — Sie geben dies an, wenn Sie die Knotengruppe erstellen oder aktualisieren.
- Erweiterte API Details/Kündigung deaktivieren (`DisableApiTermination`) — AWS PCS muss den Lebenszyklus der von ihr gestarteten Knotengruppen-Instances kontrollieren.
- Erweiterte Details/disable API stop (`DisableApiStop`) — AWS PCS muss den Lebenszyklus der von ihm gestarteten Knotengruppen-Instances kontrollieren.

- Erweiterte Details/STOP — Verhalten im Ruhezustand (**`HibernationOptions`**) — AWS PCS unterstützt den Ruhezustand von Instanzen nicht.
- Erweiterte Details/Elastic GPU (`ElasticGpuSpecifications`) — Amazon Elastic Graphics hat am 8. Januar 2024 das Ende seiner Nutzungsdauer erreicht.
- Erweiterte Details/Elastic Inference (`ElasticInferenceAccelerators`) — Amazon Elastic Inference ist für Neukunden nicht mehr verfügbar.
- AAdvancedCPUDetails/Optionen spezifizieren/Threads pro Kern (**`ThreadsPerCore`**) — AWS PCS legt die Anzahl der Threads pro Kern auf 1 fest.

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

- Benutzerdaten (`UserData`) — Diese müssen mehrteilig codiert sein. Siehe Arbeiten mit EC2 Amazon-Benutzerdaten.
- 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. Die von AMI Ihnen angegebene muss kompatibel sein mit AWS PCS. Weitere Informationen finden Sie unter "Amazon Machine Images (AMIs) für AWS PCS.
- Netzwerkeinstellungen/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.
- Netzwerkeinstellungen/Erweiterte Netzwerkkonfiguration (`NetworkInterfaces`) — Wenn Sie EC2 Instances mit einer einzigen Netzwerkkarte verwenden und keine spezielle Netzwerkkonfiguration benötigen, AWS PCS kann ich das Instanznetzwerk für Sie konfigurieren. Um mehrere Netzwerkkarten zu konfigurieren oder den Elastic Fabric Adapter auf Ihren Instances zu aktivieren, verwenden Sie. `NetworkInterfaces` IDsUnter jeder Netzwerkschnittstelle muss eine Liste der Sicherheitsgruppen enthalten sein`Groups`. Weitere Informationen finden Sie unter Mehrere Netzwerkschnittstellen in AWS PCS.
- Erweiterte Details/Kapazitätsreservierung (`CapacityReservationSpecification`) — Dies kann eingestellt werden, kann aber nicht auf eine bestimmte Angabe verweisen, `CapacityReservationId` wenn Sie damit arbeiten. AWS PCS Sie können jedoch auf eine Kapazitätsreservierungsgruppe verweisen, wenn diese Gruppe eine oder mehrere

Kapazitätsreservierungen enthält. Weitere Informationen finden Sie unter Kapazitätsreservierungen in AWS PCS.

Erstellen einer grundlegenden Startvorlage

Sie können eine Startvorlage mit dem AWS Management Console oder dem erstellen AWS CLI.

AWS Management Console

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 EC2 Instances verwendet werden soll, die von verwaltet werden AWS PCS. 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 Ihrem AWS PCS Cluster stammenVPC. 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 gestartet wurden AWS PCS
   - (Optional) Eine Sicherheitsgruppe, die eingehenden SSH Zugriff auf interaktive Instances 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. Ersetzen *region-code* mit AWS-Region dem, mit dem Sie arbeiten AWS PCS
  b. Ersetzen *my-launch-template-name* mit einem Namen für Ihre Vorlage. Es muss für das AWS-Konto und, das AWS-Region Sie verwenden, eindeutig sein.
  c. Ersetzen *my-ssh-key-name* mit dem Namen Ihres bevorzugten SSH Schlüssels.
  d. Ersetzen *sg-ExampleID1* and *sg-ExampleID2* mit einer SicherheitsgruppeIDs, die die Kommunikation zwischen Ihren EC2 Instances und dem Scheduler sowie die Kommunikation zwischen EC2 Instances ermöglicht. Wenn Sie nur über eine Sicherheitsgruppe verfügen, die den gesamten Datenverkehr ermöglicht, können Sie das vorangegangene Komma `sg-ExampleID2` und das vorangegangene Komma entfernen. Sie können auch weitere Sicherheitsgruppen IDs hinzufügen. Alle Sicherheitsgruppen, die Sie in die Startvorlage aufnehmen, müssen aus Ihrem AWS PCS Cluster stammenVPC.

```
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 EC2 Amazon-Benutzerdaten

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 registriert AWS PCSAPI, wohingegen Benutzerdatenblöcke mit dem Inhaltstyp erst 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 MIMEmehrteiligen 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 zu einer einzigen MIME mehrteiligen Datei zusammenfassen.

Eine MIME mehrteilige Datei besteht aus den folgenden Komponenten:

- Deklaration von Inhaltstyp und Teilgrenze: `Content-Type: multipart/mixed; boundary="==BOUNDARY=="`
- Die MIME Versionserklärung: `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 MIME mehrteiligen Datei signalisiert:`--==BOUNDARY==--`. Sie müssen die Linie vor der schließenden Grenze leer lassen.

> **Note**
>
> Wenn Sie Benutzerdaten zu einer Startvorlage in der EC2 Amazon-Konsole hinzufügen, können Sie sie als Klartext einfügen. Oder Sie können es aus einer Datei hochladen. Wenn Sie AWS CLI oder an verwenden AWS SDK, 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 EC2 Amazon-Benutzerdaten.

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 Skripte 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 EC2 Amazon-Benutzerdaten.

Dieses Skript 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 in diesem Skript durch Ihre eigenen Details:

- *my-bucket-name* — Der Name eines S3-Buckets, aus dem Ihr Konto lesen kann.
- *path* — Der Pfad relativ zum S3-Bucket-Root.
- *shell* — Die Linux-Shell, die zum Ausführen 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://my-bucket-name/path /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.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::my-bucket-name",
                "arn:aws:s3:::my-bucket-name/path/*"
            ]
        }
    ]
}
```

Beispiel: Setzen Sie globale Umgebungsvariablen für AWS PCS

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

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 von `"userData"` in Ihrer Startvorlage an. Weitere Informationen finden Sie unter Arbeiten mit EC2 Amazon-Benutzerdaten.

In diesem Beispiel wird das EFS Beispiel-Mount-In erweitertVerwenden von Netzwerkdateisystemen mit AWS PCS, um ein gemeinsam genutztes Home-Verzeichnis zu implementieren. Der Inhalt von /home wird gesichert, bevor das EFS Dateisystem eingehängt 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, in 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==--
```

Passwortlos aktivieren SSH

Sie können auf dem Beispiel für ein Shared Home-Verzeichnis aufbauen, um SSH Verbindungen zwischen Cluster-Instanzen 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 PCS

Sie können EC2 Amazon-Kapazität in einer bestimmten Availability Zone und für einen bestimmten Zeitraum reservieren, indem Sie On-Demand-Kapazitätsreservierungen oder EC2 Kapazitätsblöcke verwenden, um sicherzustellen, dass Sie über die erforderliche Rechenkapazität verfügen, wenn Sie sie benötigen.

> **Note**
>
> AWS PCSunterstützt On-Demand-Kapazitätsreservierungen (ODCR), unterstützt derzeit jedoch keine Kapazitätsblöcke für ML.

Wird ODCRs mit verwendet AWS PCS

Sie können wählen, wie AWS PCS Ihre Reserved Instances genutzt werden. Wenn Sie eine offene Instanz erstellenODCR, werden alle passenden Instances, die von Ihrem Konto gestartet wurden, AWS PCS oder andere Prozesse in Ihrem Konto auf die Reservierung angerechnet. Bei einem Targeting werden nur InstancesODCR, 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 Targeting verwendet, ODCR indem Sie es zu einer Startvorlage hinzufügen. Gehen Sie dazu wie folgt vor:

1. Erstellen Sie eine gezielte Kapazitätsreservierung auf Abruf (ODCR).
2. Fügen Sie die ODCR zu einer Kapazitätsreservierungsgruppe hinzu.
3. Ordnen Sie die Gruppe „Kapazitätsreservierung“ einer Startvorlage zu.
4. Erstellen oder aktualisieren Sie eine AWS PCS Rechenknotengruppe, um die Startvorlage zu verwenden.

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

Mit diesem Beispielbefehl wird ein Targeting für 32 hpc6a.48xlarge-Instances erstellt. ODCR 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 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 wird ein ARN für das Neue seinODCR. Um den Befehl ODCR with verwenden zu können AWS PCS, muss er einer Kapazitätsreservierungsgruppe hinzugefügt werden. Dies liegt daran, dass einzelne Personen AWS PCS nicht unterstützt ODCRs werden. Weitere Informationen finden Sie unter Kapazitätsreservierungsgruppen im Amazon Elastic Compute Cloud-Benutzerhandbuch.

So fügen Sie die Gruppe mit dem Namen „Kapazitätsreservierung“ ODCR zu einer Gruppe mit dem Namen hinzu`EXAMPLE-CR-GROUP`.

```
aws resource-groups group-resources --group EXAMPLE-CR-GROUP \
    --resource-arns arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/
cr-1234567890abcdef1
```

Nachdem die Gruppe ODCR erstellt und zu einer Kapazitätsreservierung hinzugefügt wurde, kann sie nun mit einer AWS PCS Rechenknotengruppe verbunden werden, indem sie 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 Rechenknotengruppe, um hpc6a.48xlarge-Instances zu verwenden, und verwenden Sie die Startvorlage, die auf die ODCR verweist, in ihrer Kapazitätsreservierungsgruppe. Legen Sie für eine statische Knotengruppe die Minimal- und Maximalanzahl der Instances auf die Größe der Reservierung fest (32). Legen Sie für eine dynamische Knotengruppe die Mindestanzahl der Instanzen auf 0 und die Höchstzahl auf die Reservierungsgröße fest.

Dieses Beispiel ist eine einfache Implementierung einer SingleODCR, die für eine Rechenknotengruppe bereitgestellt wurde. AWS PCSUnterstützt jedoch viele andere Designs. Sie können beispielsweise eine große Gruppe ODCR oder eine Kapazitätsreservierungsgruppe auf mehrere Rechenknotengruppen aufteilen. Oder Sie können das Konto verwendenODCRs, das ein anderes AWS Konto erstellt und mit Ihrem geteilt hat. Die wichtigste Einschränkung besteht darin, dass sie ODCRs immer in einer Kapazitätsreservierungsgruppe enthalten sein muss.

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

Nützliche Parameter für Startvorlagen

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

Aktivieren Sie die detaillierte CloudWatch Überwachung

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

AWS Management Console

> 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 (IMDSv2)

Die Verwendung IMDS von Version 2 mit EC2 Instances bietet erhebliche Sicherheitsverbesserungen und trägt dazu bei, potenzielle Risiken im Zusammenhang mit dem Zugriff auf Instanz-Metadaten in AWS Umgebungen zu minimieren.

AWS Management Console

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 PCSWarteschlangen

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 Job-Warteschlangen 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 erstellen in AWS PCS

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

Voraussetzungen

- Ein AWS PCS Cluster — Warteschlangen können nur in Verbindung mit einem bestimmten PCS Cluster erstellt werden.
- Eine oder mehrere AWS PCS Compute-Knotengruppen — eine Warteschlange muss mindestens einer PCS Compute-Knotengruppe zugeordnet sein.

Um eine Warteschlange zu erstellen in AWS PCS

Sie können eine Warteschlange mit dem AWS Management Console oder dem erstellen AWS CLI.

AWS Management Console

Um eine Warteschlange mit der Konsole zu erstellen

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 erstellen möchten. 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) Fügen Sie unter Tags beliebige Tags zu Ihrer AWS PCS Warteschlange hinzu
5. Wählen Sie Create queue (Warteschlange erstellen) aus. Im Statusfeld wird Creating angezeigt, während die Warteschlange eingerichtet wird. 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

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

1. Ersetzen *region-code* mit der AWS Region, in der Sie Ihren Cluster erstellen möchten.
2. Ersetzen *my-queue* mit dem Namen für Ihre Warteschlange. Der Name darf nur alphanumerische Zeichen (wobei die Groß- und Kleinschreibung beachtet werden muss) und Bindestriche enthalten. Es muss mit einem alphabetischen Zeichen beginnen und darf nicht länger als 25 Zeichen sein. Der Name muss innerhalb des Clusters eindeutig sein.
3. Ersetzen *my-cluster* mit dem Namen oder clusterId Ihres Clusters.
4. Ersetzen Sie den Wert für `computeNodeGroupId` durch Ihre eigene Compute-Knotengruppen-ID. Beachten Sie, dass Sie beim Erstellen einer Warteschlange keine Namen für Compute-Knotengruppen angeben können.

```
aws pcs create-queue --region region-code \
```

```
    --queue-name my-queue \
    --cluster-identifier my-cluster \
    --compute-node-group-configurations \
    computeNodeGroupId=computeNodeGroupExampleID1
```

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

Eine AWS PCS Warteschlange aktualisieren

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

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 Compute-Knotengruppe zu aktualisieren

Sie können eine Knotengruppe mithilfe der AWS Management Console oder der aktualisieren AWSCLI.

AWS Management Console

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 Konfiguration der Warteschlange einen der folgenden Werte:
   - Knotengruppen — Fügen Sie Compute-Knotengruppen hinzu oder entfernen Sie sie aus der Zuordnung zur Warteschlange.
   - Tags — Fügen Sie Tags für die Warteschlange hinzu oder entfernen Sie sie.
5. Wählen Sie Aktualisieren. 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. Ersetzen *region-code* mit dem AWS-Region , in dem Sie Ihren Cluster erstellen möchten.
   b. Ersetzen *my-queue* mit dem Namen oder `computeNodeGroupId` für deine Warteschlange.
   c. Ersetzen *my-cluster* mit dem 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
```

2. Die Aktualisierung 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 darüber, wie Sie eine Warteschlange in löschen 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 sie manuell mit den systemeigenen Befehlen des Schedulers zu stoppen/abzubrechen (z. B. `scancel` für Slurm).

Lösche die Warteschlange

Sie können das AWS Management Console oder verwenden AWS CLI , um eine Warteschlange zu löschen.

AWS Management Console

So löschen Sie eine Warteschlange

1. Öffnen Sie die AWS PCSKonsole.
2. Wählen Sie den Cluster der Warteschlange aus.
3. Navigieren Sie zu Warteschlangen und wählen Sie die zu löschende Warteschlange aus.
4. Wählen Sie Löschen.
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 *`region-code`* mit dem, in dem sich AWS-Region Ihr Cluster befindet.
  - Ersetzen *`my-queue`* mit dem Namen oder der ID Ihrer Warteschlange.
  - Ersetzen *`my-cluster`* mit dem Namen oder der 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 PCSLogin-Knoten

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 Instanz so konfigurieren, dass sie als Anmeldeknoten fungiert.

Themen

Verwendung einer AWS PCS Rechenknotengruppe zur Bereitstellung von Anmeldeknoten

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

Eine AWS PCS Rechenknotengruppe für Anmeldeknoten erstellen

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 Instanz in der Compute-Knotengruppe fest.
- Wählen Sie die Kaufoption auf Abruf, um zu vermeiden, dass Ihre Instanz (en) zurückgefordert werden.

- Wählen Sie einen aussagekräftigen Namen für die Compute-Knotengruppe, z. B. Login.
- Wenn Sie möchten, dass auf die Login-Node-Instanz (en) auch außerhalb Ihres eigenen zugegriffen werden kannVPC, 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 Instanzprofil sollte nur die AWS Berechtigungen haben, die Sie Ihren Endbenutzern geben möchten. Details dazu finden Sie unter IAMInstanzprofile für AWS Parallel Computing Service.
- Erwägen Sie, AWS Systems Manager Session Manager die Verwaltung Ihrer Anmeldeinstanzen zu gestatten.
- Erwägen Sie, den Zugriff auf die AWS Instanzanmeldedaten auf Administratorbenutzer zu beschränken
- Wählen Sie kostengünstigere Instanztypen als für reguläre Rechenknotengruppen aus, da die Anmeldeknoten kontinuierlich ausgeführt werden.
- Verwenden Sie dasselbe (oder ein Derivat) AMI wie für Ihre anderen Compute-Knotengruppen, um sicherzustellen, dass auf allen Instanzen dieselbe Software installiert ist. Weitere Informationen zum Anpassen finden Sie AMIs unter Amazon Machine Images (AMIs) für AWS PCS
- Konfigurieren Sie dasselbe Netzwerk-Dateisystem (AmazonEFS, 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 ACTIVE den Status erreicht hat, können Sie die EC2 Instanz (en) finden, die sie erstellt hat, und sich bei ihnen anmelden. Weitere Informationen finden Sie unter Suchen nach Instanzen der Compute-Knotengruppe in AWS PCS.

Aktualisierung einer AWS PCS Rechenknotengruppe für Anmeldeknoten

Sie können eine Anmeldeknotengruppe aktualisieren mit UpdateComputeNodeGroup. Im Rahmen der Aktualisierung der Knotengruppe 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 einer 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 EC2 Amazon-Startvorlagen mit AWS PCS.

Löschen einer AWS PCS Rechenknotengruppe für Anmeldeknoten

Sie können eine Anmeldeknotengruppe mithilfe des Mechanismus zum Löschen von Compute-Knotengruppen unter aktualisieren AWS PCS. 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.

Standalone-Instanzen als AWS PCS Login-Knoten verwenden

Sie können unabhängige EC2 Instanzen 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 Managements betrieben werden. AWS PCS Zu diesem Zweck muss jede eigenständige Instanz:

1. Eine kompatible Slurm-Softwareversion installiert haben.
2. In der Lage sein, eine Verbindung zum AWS PCS Slurmctld-Endpunkt des Clusters herzustellen.
3. Sorgen Sie dafür, dass der Slurm Auth and Cred Kiosk Daemon (`sackd`) ordnungsgemäß mit dem Endpunkt und dem geheimen Schlüssel des Clusters konfiguriert ist. AWS PCS 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 Cluster herstellt. AWS PCS

Inhalt

Schritt 1 — Rufen Sie die Adresse und das Geheimnis für den AWS PCS Zielcluster ab

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

- Ersetzen *region-code* mit dem AWS-Region Ort, an dem der Zielcluster ausgeführt wird.
- Ersetzen *cluster-ident* mit dem Namen oder der Kennung für den Zielcluster

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

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


```
{
    "cluster": {
        "name": "independent-instance-demo",
        "id": "s3431v9rx2",
        "arn": "arn:aws:pcs:us-east-1:012345678901:cluster/s3431v9rx2",
        "status": "ACTIVE",
        "createdAt": "2024-07-12T15:32:27.225136+00:00",
        "modifiedAt": "2024-07-12T15:32:27.225136+00:00",
        "scheduler": {
            "type": "SLURM",
            "version": "23.11"
        },
        "size": "SMALL",
        "networking": {
            "subnetIds": [
                "subnet-0123456789abdef"
            ],
            "securityGroupIds": [
                "sg-0123456789abdef"
            ]
        },
        "endpoints": [
            {
                "type": "SLURMCTLD",
                "privateIpAddress": "10.3.149.220",
                "port": "6817"
            }
```

```
        ],
        "authKey": {
            "secretArn": "arn:aws:secretsmanager:us-east-1:123456789012:secret:pcs!
slurm-secret-s3431v9rx2-FN7tJFf",
            "secretVersion": "ff58d1fd-070e-4bbc-98a0-64ef967cebcc"
        }
    }
}
```

In diesem Beispiel hat der Cluster-Slurm-Controller-Endpunkt die IP-Adresse `10.3.149.220` und er läuft auf dem Port6817. 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 eine EC2-Instance wie folgt:

1. Öffnen Sie die EC2Amazon-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. `PCS-LoginNode` B. Der Name wird der Instance als Ressourcen-Tag (`Name=PCS-LoginNode`) zugewiesen.
4. Wählen Sie im Abschnitt Anwendungs- und Betriebssystemimages eine AMI für eines der Betriebssysteme aus, die von unterstützt werden AWS PCS. Weitere Informationen finden Sie unter Unterstützte Betriebssysteme.
5. Wählen Sie im Abschnitt Instanztyp einen unterstützten Instanztyp 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 Edit (Bearbeiten) aus.

        i. Wählen Sie den VPC Ihres AWS PCS Clusters aus.

        ii. Wählen Sie für Firewall (Sicherheitsgruppen) die Option Bestehende Sicherheitsgruppe auswählen aus.

A. Wählen Sie eine Sicherheitsgruppe aus, die den Datenverkehr zwischen der Instanz und dem Slurm-Controller des AWS PCS Zielclusters 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 — Installieren Sie Slurm.

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
```

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

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

> **Warning**
>
> In einer Mehrbenutzerumgebung kann möglicherweise jeder Benutzer mit Zugriff auf die Instanz das Clustergeheimnis abrufen, wenn er auf den Instanz-Metadatenservice () IMDS zugreifen kann. Dies wiederum könnte es ihnen ermöglichen, sich als andere Benutzer auszugeben. Erwägen Sie, den Zugriff nur auf Root- oder Administratorbenutzer IMDS 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 Cluster AWS PCS

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

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

```
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-23.11/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 ausgeführt wird. 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 2024-07-16 16:34:55 UTC; 8s ago
 Main PID: 9985 (sackd)
   CGroup: /system.slice/sackd.service
           ##9985 /opt/aws/pcs/scheduler/slurm-23.11/sbin/sackd --systemd --conf-
server=10.3.149.220:6817

Jul 16 16:34:55 ip-10-3-27-112.ec2.internal systemd[1]: Starting Slurm auth and cred
 kiosk daemon...
Jul 16 16:34:55 ip-10-3-27-112.ec2.internal systemd[1]: Started Slurm auth and cred
 kiosk daemon.
Jul 16 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 ein Beispiel für die Ausgabe von`sinfo`.

```
[root@ip-10-3-27-112 ~]# /opt/aws/pcs/scheduler/slurm-23.11/bin/sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
all up infinite 4 idle~ compute-[1-4]
```

Sie sollten auch in der Lage sein, Jobs einzureichen. Ein Befehl, der diesem Beispiel ähnelt, würde beispielsweise einen interaktiven Job auf einem Knoten im Cluster starten.

```
/opt/aws/pcs/scheduler/slurm-23.11/bin/srun --nodes=1 -p all --pty bash -i
```

AWS PCSNetzwerkbetrieb

Ihr AWS PCS Cluster wird in einem Amazon erstelltVPC. 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 zu Startvorlagen finden Sie unterVerwenden von EC2 Amazon-Startvorlagen mit AWS PCS.

Themen

AWS PCSVPCund Subnetzanforderungen und Überlegungen

Wenn Sie einen AWS PCS Cluster erstellen, geben Sie darin VPC ein Subnetz an. VPC Dieses Thema bietet einen Überblick über AWS PCS spezifische Anforderungen und Überlegungen für die VPC Subnetze und Subnetze, die Sie mit Ihrem Cluster verwenden. Wenn Sie kein mit verwenden VPC müssen AWS PCS, können Sie eines mithilfe einer von Ihnen AWS bereitgestellten AWS CloudFormation Vorlage erstellen. Weitere Informationen zu VPCs finden Sie unter Virtuelle private Clouds (VPC) im VPCAmazon-Benutzerhandbuch.

VPCAnforderungen und Überlegungen

Wenn Sie einen Cluster erstellen, muss der VPC von Ihnen angegebene Cluster die folgenden Anforderungen und Überlegungen erfüllen:

- Sie VPC müssen 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 VPCAmazon-Benutzerhandbuch.

- Sie VPC müssen einen DNS Hostnamen haben und die DNS Auflösung unterstützen. Andernfalls können die Knoten den Kundencluster nicht registrieren. Weitere Informationen finden Sie unter DNSAttribute für Sie VPC im VPCAmazon-Benutzerhandbuch.
- VPCMöglicherweise müssen VPC Endgeräte dies verwenden AWS PrivateLink , um Kontakt mit dem AWS PCS API aufnehmen zu können. Weitere Informationen finden Sie unter Connect your VPC to services using AWS PrivateLink im VPCAmazon-Benutzerhandbuch.

Subnetz-Anforderungen und -Überlegungen

Wenn Sie einen Slurm-Cluster erstellen, AWS PCS wird ein Elastic Network Interface (ENI) in dem von Ihnen angegebenen Subnetz erstellt. Diese Netzwerkschnittstelle ermöglicht die Kommunikation zwischen dem Scheduler-Controller und dem Kunden. VPC Die Netzwerkschnittstelle ermöglicht es Slurm auch, mit den im Kundenkonto 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, die von verwendet werden kann. AWS PCS
- Das Subnetz darf sich nicht in AWS Outposts AWS Wavelength, oder einer AWS lokalen Zone befinden.
- 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, das Sie bei der Erstellung Ihres AWS PCS Clusters angeben, sowie in anderen Subnetzen im selben Subnetz bereitstellen. 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 Sie Knoten in einem öffentlichen Subnetz bereitstellen möchten, muss dieses Subnetz automatisch öffentliche Adressen zuweisenIPv4.
- Wenn es sich bei dem Subnetz, in dem Sie Knoten bereitstellen, um ein privates Subnetz handelt und die Routing-Tabelle keine Route zu einem Netzwerkadressübersetzungsgerät (NAT) () enthält, fügen Sie VPC Endpunkte hinzu, die den Kunden verwenden. IPv4 AWS PrivateLink VPC VPCEndpunkte werden für alle AWS Dienste benötigt, mit denen die Knoten Kontakt aufnehmen. Der einzige erforderliche Endpunkt besteht darin, dem Knoten AWS PCS zu ermöglichen, die `registerNodeGroupInstances` API Aktion aufzurufen.
- Der Status eines öffentlichen oder privaten Subnetzes hat keinen Einfluss AWS PCS. Die erforderlichen Endpunkte müssen erreichbar sein.

Erstellen eines VPC für Ihren AWS PCS Cluster

Sie können innerhalb von AWS Parallel Computing Service (VPC) eine Amazon Virtual Private Cloud (Amazon AWS PCS) für Ihre Cluster erstellen.

Verwenden Sie AmazonVPC, um VPC Ressourcen in einem von Ihnen definierten virtuellen Netzwerk bereitzustellen. 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 Ihnen, sich vor der Bereitstellung von VPC Produktionsclustern gründlich mit dem VPC Service von Amazon vertraut zu machen. Weitere Informationen finden Sie unter Was ist AmazonVPC? im visuellen Modus des Autors. VPCAmazon-Benutzerhandbuch.

Ein PCS Cluster, Knoten und unterstützende Ressourcen (wie Dateisysteme und Verzeichnisdienste) werden in Ihrem Amazon bereitgestelltVPC. Wenn Sie ein vorhandenes Amazon VPC mit verwenden möchtenPCS, muss es die unter beschriebenen Anforderungen erfüllenAWS PCSVPCund Subnetzanforderungen und Überlegungen. In diesem Thema wird beschrieben, wie Sie mithilfe einer AWS bereitgestellten AWS CloudFormation Vorlage eine erstellenVPC, die den PCS Anforderungen entspricht. 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 ein Amazon VPC für zu erstellenPCS, müssen Sie über die erforderlichen IAM Berechtigungen zum Erstellen von VPC Amazon-Ressourcen verfügen. Bei diesen Ressourcen handelt es sich VPCs um Subnetze, Sicherheitsgruppen, Routing-Tabellen und Routen sowie Internet und NAT Gateways. Weitere Informationen finden Sie unter Create a VPC with a public subnet im VPCAmazon-Benutzerhandbuch. Die vollständige Liste für Amazon finden Sie unter AktionenEC2, Ressourcen und Bedingungsschlüssel für Amazon EC2 in der Service Authorization Reference.

Erstelle ein Amazon VPC

Erstellen Sie ein, VPC indem Sie das URL für den Ort, an AWS-Region dem Sie es verwenden PCS möchten, geeignete kopieren und einfügen. Sie können die AWS CloudFormation Vorlage auch herunterladen und selbst auf die AWS 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
```

- Nur Vorlage

```
https://aws-hpc-recipes.s3.us-east-1.amazonaws.com/main/recipes/net/hpc_large_scale/
assets/main.yaml
```

Um ein Amazon zu erstellen VPC für PCS

1. Öffnen Sie die Vorlage in der AWS CloudFormation Konsole.

   > **Note**
   >
   > Diese 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. `hpc-networking`
3. Geben Sie unter Parameter die folgenden Details ein:

   a. Geben VPCSie dann CidrBlockunter ein `10.3.0.0/16`

   b. Unter Subnetze A:

      i. Geben Sie dann CidrPublicSubnetA ein `10.3.0.0/20`

      ii. Dann CidrPrivateSubnetA, gib ein `10.3.128.0/20`

   c. Unter Subnetze B:

      i. Geben Sie dann CidrPublicSubnetB ein `10.3.16.0/20`

      ii. Geben Sie dann CidrPrivateSubnetA 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 gesetzt ist`True`.

      ii. Geben Sie dann CidrPublicSubnetB ein `10.3.32.0/20`

      iii. Geben Sie dann CidrPrivateSubnetA ein `10.3.160.0/20`

4. Aktivieren Sie unter Funktionen das Kontrollkästchen Ich bestätige, dass dadurch IAM Ressourcen erstellt werden AWS CloudFormation könnten.

Überwachen Sie den Status des AWS CloudFormation Stacks. Wenn der Wert erreicht ist`CREATE_COMPLETE`, können Sie die VPC Ressource verwenden.

> **Note**
>
> Um alle Ressourcen zu sehen, die mit der AWS CloudFormation Vorlage erstellt wurden, öffnen Sie die AWS CloudFormation Konsole. Wählen Sie das `hpc-networking`-Stack, und wählen Sie dann die Registerkarte Ressourcen.

Sicherheitsgruppen in AWS PCS

Sicherheitsgruppen in Amazon EC2 agieren als virtuelle Firewalls, um den ein- und ausgehenden Datenverkehr zu Instances zu kontrollieren. 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 zu Startvorlagen finden Sie unterVerwenden von EC2 Amazon-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 PCSerstellt ein kontoübergreifendes Elastic Network Interface (ENI) in dem Subnetz, das Sie bei der Erstellung eines Clusters angeben. Dadurch erhält der HPC Scheduler, der in einem von verwalteten Konto ausgeführt wird AWS, einen Pfad für die Kommunikation mit EC2 Instances, die von gestartet wurden. AWS PCS Sie müssen dafür eine Sicherheitsgruppe bereitstellen, ENI die eine bidirektionale Kommunikation zwischen dem Scheduler ENI und Ihren Cluster-Instances ermöglicht. EC2

Eine einfache Möglichkeit, dies zu erreichen, besteht darin, eine permissive, selbstreferenzierende Sicherheitsgruppe zu erstellen, die TCP /IP-Verkehr auf allen Ports zwischen allen Mitgliedern der Gruppe zulässt. Sie können dies sowohl an die Cluster- als auch an die Knotengruppeninstanzen anhängen. EC2

Beispiel für eine permissive Sicherheitsgruppenkonfiguration

Regeltyp	Protokolle	Ports	Quelle	Ziel
Eingehend	Alle	Alle	Selbst	
Ausgehend	Alle	Alle		0.0.0.0/0
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 den Datenverkehr. EFA

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 externen Instanzen `slurmctld` EC2
- 6818 — aktiviert ausgehende Verbindungen von `slurmctld` zu Instances, die auf Instances ausgeführt werden `slurmd` EC2

Die mit Ihren Rechenknoten verbundene Sicherheitsgruppe muss die folgenden Ports zulassen:

- 6817 — ermöglicht ausgehende Verbindungen zu `slurmctld` externen EC2 Instanzen.
- 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`
- EFAVerkehr zwischen Knotengruppen-Instanzen. Weitere Informationen finden Sie unter Prepare an EFA -enabled Security Group im Benutzerhandbuch für Linux-Instances
- Jeder andere Datenverkehr zwischen den Knoten, der für Ihren Workload erforderlich ist

Mehrere Netzwerkschnittstellen in AWS PCS

Einige EC2 Instanzen 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 Netzwerkschnittstellen zur EC2 Startvorlage 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 die folgenden Details:

- Das Subnetz für jede Netzwerkschnittstelle muss das gleiche sein, das Sie bei der Konfiguration der AWS PCS Rechenknotengruppe ausgewählt haben, die die Startvorlage verwendet.
- Das primäre Netzwerkgerät, auf dem die routinemäßige Netzwerkkommunikation wie SSH der HTTPS Datenverkehr stattfindet, wird durch die Einstellung von festgelegt. `DeviceIndex` `0` Andere Netzwerkschnittstellen haben den 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 SSH Sicherheitsgruppe (`sg-`*`SshSecurityGroupId`*) für eingehenden Datenverkehr 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 Instanzen in AWS PCS

Sie können eine Platzierungsgruppe verwenden, um die Platzierung von EC2 Instances so zu beeinflussen, dass sie den Anforderungen des Workloads entspricht, der 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 EC2 Amazon-Instances im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Wir empfehlen Ihnen, eine Cluster-Platzierungsgruppe einzubeziehen, wenn Sie eine AWS PCS Compute-Knotengruppe für die Verwendung des Elastic Fabric Adapters (EFA) konfigurieren.

Um eine Cluster-Platzierungsgruppe zu erstellen, die funktioniert mit EFA

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 aus dem Folgenden herunter URL 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 Adapters (EFA) mit AWS PCS

Der Elastic Fabric Adapter (EFA) ist eine leistungsstarke, fortschrittliche Netzwerkverbindung AWS , die Sie mit Ihrer EC2 Instance verbinden können, um High Performance Computing (HPC) und Machine-Learning-Anwendungen zu beschleunigen. Um Ihre Anwendungen zu aktivieren, die auf einem AWS PCS Cluster ausgeführt EFA werden, müssen Sie die AWS PCS Compute-Knotengruppen-Instances so konfigurieren, dass sie EFA wie folgt verwendet werden.

Inhalt

Installieren Sie EFA auf einem AWS PCS -kompatiblen AMI

Auf der in der AWS PCS Compute-Knotengruppe AMI verwendeten Gruppe muss der EFA Treiber installiert und geladen sein. Informationen zum Erstellen eines benutzerdefinierten Systems AMI mit installierter EFA Software finden Sie unterBenutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS.

Identifizieren Sie EFA -aktivierte Instanzen EC2

Zur Verwendung EFA müssen alle Instanztypen, die für eine AWS PCS Compute-Gruppe zugelassen sindEFA, Unterstützung bieten und dieselbe Anzahl von vCPUs (und GPUs falls zutreffend) aufweisen. Eine Liste der EFA -fähigen 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 unterstützten Instance-Typen einzusehen. EFA Ersetzen *region-code* mit dem AWS-Region Ort, den Sie verwenden AWS PCS, z. `us-east-1` B.

```
aws ec2 describe-instance-types \
   --region region-code \
   --filters Name=network-info.efa-supported,Values=true \
   --query "InstanceTypes[*].[InstanceType]" \
   --output text | sort
```

Ermitteln Sie, wie viele Netzwerkschnittstellen verfügbar sind

Einige EC2 Instanzen haben mehrere Netzwerkkarten. Dadurch können sie mehrere habenEFAs. 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 Folgendes unterstütztEFA: Der Befehl gibt eine Sicherheitsgruppen-ID aus. Nehmen Sie die folgenden Ersetzungen vor:

- *region-code*— Geben Sie an AWS-Region , wo Sie AWS PCS es verwenden, z. B. `us-east-1`
- *vpc-id*— Geben Sie die ID der anVPC, für die Sie verwenden AWS PCS.
- *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 Folgendes unterstütztEFA. Laden Sie die Vorlage von der folgenden Seite herunter URL 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 VPCunter den VPC Ort aus, den Sie verwenden möchten AWS PCS.

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

Es wird empfohlen, alle Instances zu starten, die EFA in einer Cluster-Platzierungsgruppe verwendet werden, um die physische Entfernung zwischen ihnen zu minimieren. Wir empfehlen, für jede Rechenknotengruppe, die Sie verwenden möchten, eine Platzierungsgruppe zu erstellenEFA. Informationen Platzierungsgruppen für EC2 Instanzen in AWS PCS zum Erstellen einer Platzierungsgruppe für Ihre Compute-Knotengruppe finden Sie unter.

Erstellen oder aktualisieren Sie eine EC2 Startvorlage

EFANetzwerkschnittstellen werden in der EC2 Startvorlage für eine AWS PCS Rechenknotengruppe eingerichtet. Wenn mehrere Netzwerkkarten vorhanden sind, EFAs können mehrere 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. Jeder EFA benötigt die Sicherheitsgruppe, die den EFA Verkehr aktiviert (`sg-`*`EfaSecGroupId`*). Andere Sicherheitsgruppen, insbesondere solche, die normalen Datenverkehr wie SSH oder verarbeitenHTTPS, müssen nur an die primäre Netzwerkschnittstelle (gekennzeichnet durch ein `DeviceIndex` of`0`) angehängt werden. Startvorlagen, in denen Netzwerkschnittstellen definiert sind, unterstützen die Einstellung von Sicherheitsgruppen mithilfe des `SecurityGroupIds` Parameters nicht. Sie müssen für jede Netzwerkschnittstelle, die Sie `Groups` konfigurieren, einen Wert 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 die Compute-Knotengruppe

Erstellen oder aktualisieren Sie eine AWS PCS Rechenknotengruppe mit InstanzenvCPUs, die dieselbe Anzahl und dieselbe Prozessorarchitektur haben und die alle unterstützenEFA. Konfigurieren Sie die Compute-Knotengruppe so, dass sie AMI mit der darauf installierten EFA Software verwendet wird und dass sie die Startvorlage verwendet, mit der Netzwerkschnittstellen eingerichtet werden, die EFA -fähige Netzwerkschnittstellen einrichtet.

(Optional) Testen EFA

Sie können nachweisen, dass die Kommunikation zwischen zwei Knoten in einer EFA Compute-Knotengruppe aktiviert ist, indem Sie das `fi_pingpong` Programm ausführen, das in der EFA Softwareinstallation enthalten ist. Wenn dieser Test erfolgreich ist, ist er wahrscheinlich EFA richtig konfiguriert.

Zu Beginn benötigen Sie zwei laufende Instanzen 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 namens `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=23.11.8
   OS=Linux 5.10.218-208.862.amzn2.aarch64 #1 SMP Tue Jun 4 16:52:10 UTC 2024
   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=2024-07-02T19:00:09 SlurmdStartTime=2024-07-08T19:33:25
   LastBusyTime=2024-07-08T19:33:25 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@2024-07-02T18:59:00]
   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=23.11.8
   OS=Linux 5.10.218-208.862.amzn2.aarch64 #1 SMP Tue Jun 4 16:52:10 UTC 2024
   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=2024-07-02T19:00:09 SlurmdStartTime=2024-07-08T19:33:25
   LastBusyTime=2024-07-08T19:33:25 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@2024-07-02T18:59:00]
   InstanceId=i-0a2c82623cb1393a7 InstanceType=hpc7g.16xlarge
```

Stellen Sie mit (oderhpc7g-1) eine Connect zu einem der Knoten her SSH (in diesem BeispielfallSSM). Beachten Sie, dass es sich um eine interne IP-Adresse handelt, sodass Sie

möglicherweise von einem Ihrer Anmeldeknoten aus eine Verbindung herstellen müssen, wenn Sie dies verwendenSSH. 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 Startvorlage mit EFA aktivierter Option zu erstellen

Da bei der Einrichtung mehrere Abhängigkeiten bestehenEFA, 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 aus dem Folgenden herunter URL und laden Sie sie dann auf die CloudFormation Konsole hoch, AWS-Region in der Sie sie verwenden AWS PCS.

```
https://aws-hpc-recipes.s3.amazonaws.com/main/recipes/pcs/enable_efa/assets/pcs-lt-efa.yaml
```

Geben Sie bei geöffneter Vorlage in der AWS 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 VpcIdden VPC Ort aus, an dem Ihr AWS PCS Cluster bereitgestellt wird.
  - Wählen Sie NodeGroupSubnetIdunter das Subnetz in Ihrem Cluster aus, in VPC dem EFA -fähige Instances gestartet werden sollen.
  - 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 zu gewähren AWS PCSAPI. Viele Kunden wählen die Standardsicherheitsgruppe aus ihrem ClusterVPC.
  - 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 aussagekräftigen Namen für die Startvorlage ein, z. B. *NODEGROUPNAME*`-efa-lt` Der Name muss für Sie AWS-Konto in dem Bereich, den Sie verwenden AWS PCS möchten AWS-Region , eindeutig sein.
- Unter Fähigkeiten
  - Markieren Sie das Kästchen Ich bestätige, dass AWS CloudFormation dadurch IAM Ressourcen entstehen 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 die Compute-Knotengruppe.

Verwenden von Netzwerkdateisystemen mit AWS PCS

Sie können Netzwerkspeichervolumes an Knoten anhängen, die in einer Rechenknotengruppe des AWS Parallel Computing Service (AWS PCS) gestartet wurden, um einen dauerhaften Speicherort bereitzustellen, an dem Daten und Dateien geschrieben und abgerufen werden können. Sie können Volumes verwenden, die von AWS Diensten bereitgestellt werden. Zu den Volumes gehören Amazon Elastic File System (AmazonEFS) NetApp ONTAP, Amazon FSx FSx for, Amazon for Open ZFS, Amazon FSx for Lustre und Amazon File Cache. Sie können auch selbstverwaltete Volumes 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 die Compute-Knotengruppe aufgenommen wird AMI oder indem ein Skript verwendet wird, das beim Start der Instance ausgeführt wird.
- Es muss eine Netzwerkroute zwischen dem gemeinsam genutzten Speichervolume und den Instanzen der Compute-Knotengruppe bestehen.
- Die Sicherheitsgruppenregeln sowohl auf dem gemeinsam genutzten Speichervolume als auch auf den Instanzen der Compute-Knotengruppe 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 EC2 Amazon-Startvorlagen mit AWS PCS.

Beispiele für Netzwerk-Mounts

Sie können Dateisysteme mit AmazonEFS, Amazon FSx for Lustre, Amazon FSx for Open ZFS 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 EFS Amazon-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 EFS Amazon-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 mounten werden EFS
- *`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 in VPC dem Verzeichnis, das Sie verwenden AWS PCS möchten, ein Dateisystem FSx für Lustre. 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 Open ZFS

Einrichtung des Dateisystems

Erstellen Sie ein ZFS Dateisystem FSx für Open in VPC dem Verzeichnis, das Sie verwenden möchten AWS PCS. 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 AWS PCS Compute-Knotengruppen-Instances aus ermöglicht. Weitere Informationen zu Sicherheitsgruppen finden Sie unter Verwaltung des Dateisystemzugriffs mit Amazon VPC im FSxfor ZFS Open-Benutzerhandbuch.

Startvorlage

Fügen Sie Benutzerdaten hinzu, die `cloud-config` zum Mounten des Root-Volumes für ein FSx for ZFS Open-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 Ihre FSx for ZFS Open-Freigabe mounten möchten
- *`filesystem-id`*— Die Dateisystem-ID für das FSx for ZFS Open-Dateisystem
- *`region-code`*— Der AWS-Region Ort, an dem das FSx for ZFS Open-Dateisystem bereitgestellt wird (muss mit Ihrem AWS PCS System identisch sein)

```
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 demVPC, wo Sie ihn verwenden werden AWS PCS. Um Übertragungen zwischen Zonen zu minimieren, wählen Sie ein Subnetz in derselben Availability Zone aus, in der Sie die meisten Ihrer PCS Compute Node Group-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 Name des Domain Name System (DNS) für den Datei-Cache
- *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 PCS

AWS PCSarbeitet mit demAMIs, was Sie bereitstellen, und bietet eine große Flexibilität bei der Software und Konfiguration auf den Knoten in Ihrem Cluster. Wenn Sie es ausprobieren AWS PCS, können Sie ein Beispiel verwenden, das von AMI bereitgestellt und verwaltet AWS wird. Wenn Sie es AWS PCS in der Produktion verwenden, empfehlen wir Ihnen, Ihr eigenes zu erstellenAMIs. In diesem Thema erfahren SieAMIs, wie Sie das Beispiel entdecken und verwenden und wie Sie Ihr eigenes benutzerdefiniertes Beispiel erstellen und verwenden könnenAMIs.

Themen

- Verwenden von Amazon Machine Images (AMIs) als Beispiel mit AWS PCS
- Benutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS
- Softwareinstallationsprogramme, für die Sie individuell AMIs erstellen können AWS PCS

Verwenden von Amazon Machine Images (AMIs) als Beispiel mit AWS PCS

AWSstellt ein Beispiel bereitAMIs, das Sie als Ausgangspunkt für die Arbeit verwenden können AWS PCS.

> **Important**
>
> AMIsDie Beispiele dienen zu Demonstrationszwecken und werden nicht für Produktionsworkloads empfohlen.

Finden Sie das aktuelle Beispiel AWS PCS AMIs

AWS Management Console

AWSPCSBeispiele AMIs haben die folgende Benennungskonvention:

```
aws-pcs-sample_ami-OS-architecture-schdeulder-scheduler-major-version
```

Akzeptierte Werte

- *OS* – amzn2
- *architecture* — x86_64 oder arm64
- *scheduler* – slurm
- *scheduler-major-version* – 23.11

Um ein AWS PCS Beispiel zu finden AMIs

1. Öffnen Sie die EC2Amazon-Konsole.
2. Navigieren Sie zu AMIs.
3. Wählen Sie Öffentliche Abbilder aus.
4. Suchen Sie AMI unter „Nach Attribut oder Tag suchen“ AMI anhand des Vorlagennamens nach einem.

   Beispiele

   - Slurm 23.11 unterstützt Graviton AMI

     ```
     aws-pcs-sample_ami-amzn2-arm64-slurm-23.11
     ```

   - Beispiel für x86-Instanzen AMI

     ```
     aws-pcs-sample_ami-amzn2-x86_64-slurm-23.11
     ```

   > **Note**
   >
   > Wenn es mehrere gibtAMIs, verwenden Sie den AMI mit dem aktuellsten Zeitstempel.

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. Ersetzen *region-code* mit dem AWS-Region , wo Sie AWS PCS es verwenden, z. us-east-1 B.

- x86_64

```
aws ec2 describe-images --region region-code --owners amazon 533267220047
 654654292779 654654317195 975050324343 \
--filters 'Name=name,Values=aws-pcs-sample_ami-amzn2-x86_64-slurm-23.11*' \
            '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 533267220047
 654654292779 654654317195 975050324343 \
--filters 'Name=name,Values=aws-pcs-sample_ami-amzn2-arm64-slurm-23.11*' \
            'Name=state,Values=available' \
--query 'sort_by(Images, &CreationDate)[-1].[Name,ImageId]' --output text
```

Verwenden Sie die AMI ID, wenn Sie eine Rechenknotengruppe erstellen oder aktualisieren.

Erfahren Sie mehr über AWS PCS Sample AMIs

Den Inhalt und die Konfigurationsdetails für aktuelle und frühere Versionen des AWS PCS AMIs Beispiels finden Sie unterVersionshinweise für ein AWS PCS Beispiel AMIs.

Erstellen Sie Ihre eigene, AMIs kompatibel mit AWS PCS

Informationen dazu, wie Sie eigene erstellenAMIs, die mit funktionieren AWS PCS, finden Sie unterBenutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS.

Benutzerdefinierte Amazon Machine Images (AMIs) für AWS PCS

AWS PCSist so konzipiert, dass es mit Amazon Machine Images (AMI) funktioniert, die Sie in den Service einbringen. Auf diesen AMIs können beliebige Software und Konfigurationen installiert sein, sofern 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 Ihrer eigenen Festplatte zu installieren. AMI Wir empfehlen Ihnen, zur Installation AWS von Slurm bereitgestellte Installationsprogramme zu verwenden, AMI aber Sie können Slurm auch selbst installieren, wenn Sie dies bevorzugen (nicht empfohlen).

> **Note**
>
> Wenn Sie es versuchen möchten, AWS PCS ohne ein benutzerdefiniertes System zu erstellen, können Sie ein Beispiel verwendenAMI, das von bereitgestellt wird. AMI AWS Weitere Informationen finden Sie unter Verwenden von Amazon Machine Images (AMIs) als Beispiel mit AWS PCS.

Dieses Tutorial hilft Ihnen dabeiAMI, eine zu erstellen, die mit PCS Compute-Knotengruppen verwendet werden kann, um Ihre HPC und KI/ML-Workloads zu unterstützen.

Themen

Schritt 1 — Starten Sie eine temporäre Instanz

Starten Sie eine temporäre Instanz, mit der Sie die AWS PCS Software und den Slurm-Scheduler installieren und konfigurieren können. Sie verwenden diese Instanz, um eine AMI kompatible Instanz zu erstellen. AWS PCS

So starten Sie eine temporäre Instance

1. Öffnen Sie die EC2Amazon-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 Abschnitt Anwendungs- und Betriebssystemimages eines 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 gestarteten Instanzen AWS PCS für die Verwendung mit Slurm konfiguriert.

So installieren Sie den AWS PCS-Agenten

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, RHEL 9, Rocky Linux 9

      ```
      sudo yum update -y
      ```

    - Ubuntu 22.04

      ```
      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. Ersetze *region* durch den AWS-Region Ort, an dem Sie Ihre temporäre Instance gestartet haben, z. `us-east-1` B.

```
curl https://aws-pcs-repo-region.s3.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.0.0-1.tar.gz -o aws-pcs-agent-v1.0.0-1.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 kann sich in future Versionen der AWS PCS Agentsoftware 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 herunter AWS PCS und importieren Sie ihn in Ihren Schlüsselbund. Ersetze *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.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 AWS PCS Agenteninstallationsskript 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. Ersetzen *region* mit dem AWS-Region Ort, an dem Sie Ihre temporäre Instance gestartet haben, z. B. `us-east-1`

```
wget https://aws-pcs-repo-region.s3.amazonaws.com/aws-pcs-agent/aws-pcs-agent-
v1.0.0-1.tar.gz.sig && \
    gpg --verify ./aws-pcs-agent-v1.0.0-1.tar.gz.sig
```

Die Ausgabe sollte folgendermaßen oder ähnlich aussehen:

```
gpg: assuming signed data in './aws-pcs-agent-v1.0.0-1.tar.gz'
gpg: Signature made Thu Aug  8 18:50:19 2024 CEST
gpg:                using RSA key 4BAA531875430EB0739E6D961BA7F0AF6E34C496
gpg: Good signature from "AWS PCS Packages (AWS PCS Packages)" [unknown]
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: 4BAA 5318 7543 0EB0 739E  6D96 1BA7 F0AF 6E34 C496
```

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-v1.0.0-1.tar.gz && \
    cd aws-pcs-agent
```

7. Installieren Sie die AWS PCS-Software.

```
sudo ./installer.sh
```

8. Überprüfen Sie die AWS PCS Softwareversionsdatei, um zu bestätigen, dass die Installation erfolgreich war.

```
cat /opt/aws/pcs/version
```

Die Ausgabe sollte folgendermaßen oder ähnlich aussehen:

```
AGENT_INSTALL_DATE='Mon Aug 12 12:28:43 UTC 2024'
AGENT_VERSION='1.0.0'
AGENT_RELEASE='1'
```

Schritt 3 — Installieren Sie Slurm

Installieren Sie eine Version von Slurm, die kompatibel ist mit. AWS PCS

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. Ersetze *`region`* mit AWS-Region der Ihrer temporären Instanz, wie zum Beispiel`us-east-1`.

```
curl https://aws-pcs-repo-region.s3.amazonaws.com/aws-pcs-slurm/aws-pcs-
slurm-23.11-installer-23.11.9-1.tar.gz \
    -o aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz
```

Sie können die neueste Version auch abrufen, indem Sie die Versionsnummer durch `latest` den vorherigen Befehl ersetzen (zum Beispiel:`aws-pcs-slurm-23.11-installer-latest.tar.gz`).

> **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 herunter AWS PCS und importieren Sie ihn in Ihren Schlüsselbund. Ersetze *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.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. Ersetzen *region* mit dem AWS-Region Ort, an dem Sie Ihre temporäre Instanz gestartet haben, z. B. `us-east-1`

   ```
   wget https://aws-pcs-repo-region.s3.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz.sig && \
   ```

```
    gpg --verify ./aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz.sig
```

Die Ausgabe sollte folgendermaßen oder ähnlich aussehen:

```
gpg: assuming signed data in './aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz'
gpg: Signature made Thu Aug  8 14:23:38 2024 CEST
gpg:                using RSA key 4BAA531875430EB0739E6D961BA7F0AF6E34C496
gpg: Good signature from "AWS PCS Packages (AWS PCS Packages)" [unknown]
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: 4BAA 5318 7543 0EB0 739E  6D96 1BA7 F0AF 6E34 C496
```

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-23.11-installer-23.11.9-1.tar.gz && \
    cd aws-pcs-slurm-23.11-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-23.11/version
```

Die Ausgabe sollte folgendermaßen oder ähnlich aussehen:

```
SLURM_INSTALL_DATE='Mon Aug 12 12:38:56 UTC 2024'
SLURM_VERSION='23.11.9'
PCS_SLURM_RELEASE='1'
```

Schritt 4 — (Optional) Installieren Sie zusätzliche Treiber, Bibliotheken und Anwendungssoftware

Installieren Sie zusätzliche Treiber, Bibliotheken und Anwendungssoftware auf der temporären Instanz. Die Installationsverfahren variieren je nach den spezifischen Anwendungen und Bibliotheken. Wenn Sie noch keine benutzerdefinierte AMI Version erstellt haben, empfehlen wir Ihnen, zunächst nur die AWS PCS Software und Slurm zu installieren und zu testen und dann schrittweise Ihre eigene Software und Konfigurationen hinzuzufügen, sobald Sie den ersten Erfolg bestätigt haben. AWS PCS AMI

Beispiele

- Elastic Fabric Adapter (EFA) -Software. 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 (AmazonEFS). Weitere Informationen finden Sie unter Manuelles Installieren des EFS Amazon-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 FSxfor 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
- NVIDIATreiber, und CUDADCGM, um die Instanztypen p* oder g* zu verwenden.

Schritt 5 — Erstellen Sie eine kompatible AMI AWS PCS

Nachdem Sie die erforderlichen Softwarekomponenten installiert haben, erstellen Sie eine, AMI die Sie wiederverwenden können, um Instanzen in AWS PCS Compute-Knotengruppen zu starten.

Um eine AMI aus Ihrer temporären Instanz zu erstellen

1. Öffnen Sie die EC2Amazon-Konsole.
2. Wählen Sie im Navigationsbereich Instances aus.
3. Wählen Sie die temporäre Instance 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 Bildname einen beschreibenden Namen für das AMI ein.
   b. (Optional) Geben Sie unter Bildbeschreibung eine kurze Beschreibung des Zwecks von ein. AMI
   c. Wählen Sie Create Image (Image erstellen) aus.
5. Wählen Sie im Navigationsbereich AMIs.
6. Suchen Sie die AMI Datei, die Sie in der Liste erstellt haben. Warten Sie, bis sich der Status von Ausstehend auf Verfügbar ändert, und verwenden Sie ihn dann mit einer AWS PCS Compute-Knotengruppe.

Schritt 6 — Verwenden Sie den benutzerdefinierten Wert AMI mit einer AWS PCS Compute-Knotengruppe

Sie können Ihren benutzerdefinierten Wert AMI mit einer neuen oder vorhandenen AWS PCS Compute-Knotengruppe verwenden.

New compute node group

Um das benutzerdefinierte zu verwenden AMI

1. Öffnen Sie die AWS PCSKonsole.
2. Klicken Sie im Navigationsbereich auf Cluster.
3. Wählen Sie den Cluster aus, in dem Sie den benutzerdefinierten Cluster verwenden möchtenAMI, und wählen Sie dann Knotengruppen berechnen aus.
4. Erstellen Sie eine neue Compute-Knotengruppe. Weitere Informationen finden Sie unter Erstellen einer Compute-Knotengruppe in AWS PCS. Suchen Sie unter AMIID nach dem Namen oder der ID des benutzerdefinierten Objekts, das AMI Sie verwenden möchten. Schließen Sie die Konfiguration der Compute-Knotengruppe ab und wählen Sie dann Create Compute-Knotengruppe aus.

5. (Optional) Bestätigen Sie, dass die AMI unterstützten Instances gestartet werden. Starten Sie eine Instanz 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 EC2 Amazon-Konsole, bis eine Instance angezeigt wird, die mit der neuen Compute-Knotengruppen-ID gekennzeichnet ist. Weitere Informationen dazu finden Sie unterSuchen nach Instanzen der Compute-Knotengruppe in AWS PCS..

   b. Wenn Sie sehen, dass eine Instance gestartet wird und ihr Bootstrap-Vorgang abgeschlossen ist, vergewissern Sie sich, dass sie den erwarteten AMI Wert verwendet. Wählen Sie dazu die Instance aus und überprüfen Sie dann die AMIID unter Details. Sie sollte mit den Einstellungen übereinstimmen, die AMI 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 zu verwenden AMI

1. Öffnen Sie die AWS PCSKonsole.
2. Klicken Sie im Navigationsbereich auf Cluster.
3. Wählen Sie den Cluster aus, in dem Sie den benutzerdefinierten Cluster verwenden möchtenAMI, und wählen Sie dann Knotengruppen berechnen aus.
4. Wählen Sie die Knotengruppe aus, die Sie konfigurieren möchten, und klicken Sie auf Bearbeiten. Suchen Sie unter AMIID nach dem Namen oder der ID des benutzerdefinierten Objekts, das AMI Sie verwenden möchten. Beenden Sie die Konfiguration der Compute-Knotengruppe und wählen Sie dann Update aus. Neue Instanzen, die in der Compute-Knotengruppe gestartet werden, verwenden die aktualisierte AMI ID. Bestehende Instanzen werden weiterhin die alten Instanzen verwenden, AMI bis sie AWS PCS ersetzt werden. Weitere Informationen finden Sie unter Aktualisierung einer AWS PCS Compute-Knotengruppe.
5. (Optional) Vergewissern Sie sich, dass die AMI unterstützten Instances gestartet werden. Starten Sie eine Instanz 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 EC2 Amazon-Konsole, bis eine Instance angezeigt wird, die mit der neuen Compute-Knotengruppen-ID gekennzeichnet ist. Weitere Informationen dazu finden Sie unterSuchen nach Instanzen der Compute-Knotengruppe in AWS PCS..

b. Wenn Sie sehen, dass eine Instance gestartet wird und ihr Bootstrap-Vorgang abgeschlossen ist, vergewissern Sie sich, dass sie den erwarteten AMI Wert verwendet. Wählen Sie dazu die Instance aus und überprüfen Sie dann die AMIID unter Details. Sie sollte mit den Einstellungen übereinstimmen, die AMI 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 Sie wie vorgesehen AMI funktionieren AWS PCS, können Sie die temporäre Instance beenden, damit keine Gebühren mehr für sie anfallen.

So beenden Sie die temporäre Instance:

1. Öffnen Sie die EC2Amazon-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, für die Sie individuell AMIs erstellen können AWS PCS

AWS stellt eine herunterladbare Datei bereit, mit der die AWS PCS Software auf einer Instanz 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 Dateien AMIs für die Verwendung mit zu erstellen, AWS PCS oder Sie können Ihre eigenen Methoden verwenden.

Inhalt

AWS PCSSoftware-Installationsprogramm

Das AWS PCS Softwareinstallationsprogramm konfiguriert eine Instanz so, dass sie AWS PCS während des Instanz-Bootstrap-Vorgangs verwendet werden kann. Sie müssen die von Ihnen AWS bereitgestellten Installationsprogramme verwenden, um die AWS PCS Software auf Ihrem benutzerdefinierten System zu installieren. AMI

Slurm-Installationsprogramm

Das Slurm-Installationsprogramm lädt relevante Versionen von Slurm und seinen Abhängigkeiten herunter, kompiliert und installiert sie. Sie können den Slurm-Installer verwenden, um benutzerdefinierte Versionen für zu erstellen. AMIs AWS PCS Sie können auch Ihre eigenen Mechanismen verwenden, sofern diese mit der Softwarekonfiguration übereinstimmen, die der Slurm-Installer bereitstellt.

Die AWS mitgelieferte Software installiert Folgendes:

- Verwenden Sie die angeforderte Haupt- und Wartungsversion (derzeit Version 23.11.8) — Lizenz 2 GPL
  - Slurm wurde mit der Einstellung auf gebaut `--sysconfdir /etc/slurm`
  - Slurm wird mit der Option gebaut und `--enable-pam --without-munge`
  - Slurm wurde mit der Option gebaut `--sharedstatedir=/run/slurm/`
  - Slurm wurde mit und mit PMIX Support entwickelt JWT
  - Slurm ist installiert unter `/opt/aws/pcs/schedulers/slurm-23.11`
- Öffnen PMIX (Version 4.2.6) — Lizenz
  - Open PMIX ist als Unterverzeichnis installiert von `/opt/aws/pcs/scheduler/`
- libjwt (Version 1.15.3) — Lizenz -2.0 MPL

- 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-23.11.service`
- Falls sie nicht existieren, werden ein Slurm-Benutzer und eine Gruppe (`slurm:slurm`) mitUID/GIDof erstellt. `401`
- 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.
- Bei RHEL9 der Installation wird das Programm aktiviert `codeready-builder-for-rhel-9-rhui-rpms` und `epel-release-latest-9` von `fedoraproject` dort aus installiert, um Slurm oder seine Abhängigkeiten zu erstellen.

Unterstützte Betriebssysteme

Die AWS PCS Software und die Slurm-Installationsprogramme unterstützen die folgenden Betriebssysteme:

- Amazon Linux 2
- RedHat Linux für Unternehmen 9
- Rocky Linux 9
- Ubuntu 22.04

> **Note**
>
> AWS Deep Learning AMIs (DLAMI) Versionen, die auf Amazon Linux 2 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 Entwicklerhandbuch.AWS Deep Learning AMIs

Unterstützte Instance-Typen

AWS PCSSoftware- und Slurm-Installationsprogramme unterstützen jeden x86_64- oder arm64-Instanztyp, auf dem eines der unterstützten Betriebssysteme ausgeführt werden kann.

Unterstützte Slurm-Versionen

Die folgenden Hauptversionen von Slurm werden unterstützt:

- Slurm 23.11

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 AWS Support.

Der folgende Befehl generiert beispielsweise die SHA256 Prüfsumme für den Slurm 23.11.9-Tarball.

```
$ sha256sum aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz
```

Beispielausgabe:

```
1de7d919c8632fe8e2806611bed4fde1005a4fadc795412456e935c7bba2a9b8 aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz
```

In der folgenden Tabelle sind die Prüfsummen für die neuesten Versionen der Installationsprogramme aufgeführt. Ersetzen *us-east-1* mit dem, AWS-Region wo Sie es verwenden. AWS PCS

Installer (Installationsprogramm)	Herunterladen URL	SHA256Prüfsumme
Slurm 23.11.9	`https://aws-pcs-repo- us-east-1 .s3.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz`	`1de7d919c8632fe8e2806611bed4fde1005a4fadc795412456e935c7bba2a9b8`
AWS PCSAgent 1.0.0	`https://aws-pcs-repo- us-east-1 .s3.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.0.0-1.tar.gz`	`d2d3d68d00c685435c38af471d7e2492dde5ce9eb222d7b6ef0042144b134ce0`

Slurm-Versionen in AWS PCS

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 PCSunterstützt zunächst Slurm 23.11. Sie können Ihre Slurm-Hauptversion aktualisieren, nachdem eine neue Version veröffentlicht wurde. AWS PCSist so konzipiert, dass der Slurm-Controller automatisch mit Patch-Versionen aktualisiert wird.

Wenn SchedMD die Unterstützung für eine bestimmte Hauptversion beendet, beendet dies AWS PCS auch die Unterstützung für diese Hauptversion. AWS PCSsendet im Voraus eine Benachrichtigung, wenn eine Slurm-Hauptversion kurz vor dem Ende ihrer Lebensdauer 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.

Häufig gestellte Fragen zu Slurm-Versionen

Wie lange wird eine Slurm-Version AWS PCS unterstützt?

AWS PCSfolgt den SchedMD-Supportzyklen für Hauptversionen. AWS PCSunterstützt bis zu 3 Hauptversionen gleichzeitig. Nachdem SchedMD eine neue Hauptversion veröffentlicht hat, wird die AWS PCS älteste unterstützte Version eingestellt. AWS PCSveröffentlicht so bald wie möglich eine neue Hauptversion von Slurm, aber es kann zu Verzögerungen zwischen der Veröffentlichung von SchedMD und ihrer Verfügbarkeit in kommen. AWS PCS

Wann AWS PCS informiere ich mich über das Ende der Support-Laufzeit (EOSL) für Slurm-Versionen?

AWS PCSbenachrichtigt Sie mehrmals, in einem vorher festgelegten Rhythmus, vor dem Datum. EOSL

Was muss ich tun, wenn sich eine Slurm-Version nähert? EOSL

Sie müssen Ihre Slurm-Versionen vorher aktualisierenEOSL, um eine sichere und unterstützte Umgebung aufrechtzuerhalten.

Wie kann ich meine Cluster aktualisieren, um eine neue Hauptversion von Slurm zu verwenden?

Um die Slurm-Version zu aktualisieren, müssen Sie einen neuen Cluster erstellen. Sie müssen auch auf die entsprechende AWS PCS Software in Ihrem System aktualisieren AMI und diese verwenden, um die Rechenknotengruppen für Ihren neuen Cluster zu erstellen.

Wie erhalten meine Cluster neue Slurm-Patch-Versionen?

AWS PCSist so konzipiert, dass es automatisch Patches einspielt, um die häufigsten Sicherheitslücken und Sicherheitslücken in Slurm zu beheben (). CVEs AWS PCSwendet die Patches auf Cluster-Controller an, die unter internen Dienstkonten ausgeführt werden. Sie müssen die AWS PCS API Aktionen AWS Management Console oder verwenden, um Patches auf Ihren EC2 AWS-Konto Instanzen zu installieren.

Was ist, wenn ich Slurm nicht bis zum angegebenen EOSL Datum aktualisiere?

AWS PCSwurde entwickelt, um Cluster zu stoppen, die eine nicht unterstützte Slurm-Version haben. Sie müssen die Slurm-Hauptversion des Cluster-Controllers und die auf den AWS PCS Compute-Knotengruppen installierte Software aktualisieren.

Wie viele Slurm-Versionen werden unterstützt? AWS PCS

AWS PCSunterstützt bis zu 3 große Slurm-Versionen gleichzeitig, einschließlich der aktuellen und der 2 vorherigen Hauptversionen.

Welche Slurm-Versionsupdates sollte ich anwenden?

Wir empfehlen Ihnen dringend, dieselbe Hauptversion für alle Komponenten in Ihrem Cluster zu verwenden und die neuesten Patches zu installieren, sobald sie veröffentlicht werden. Die Knotengruppen AMIs für Ihre Datenverarbeitung müssen eine Version der Slurm-Software verwenden, die mit der Slurm-Version des Cluster-Controllers kompatibel ist. Die Slurm-Hauptversion in Ihrer AMIs muss sich innerhalb von 2 Versionen der Slurm-Hauptversion auf dem Cluster-Controller befinden. Die Slurm-Version, die in AMI und auf den laufenden EC2 Instances im Cluster installiert ist, darf nicht neuer sein als die Slurm-Version auf dem Cluster-Controller. Um die Unterstützung für Ihren Cluster aufrechtzuerhalten, AMIs müssen Sie eine unterstützte AWS PCS Softwareversion verwenden.

Was ist, wenn ich die Slurm-Hauptversion aktualisiere, aber ältere Slurm-Software in meinen AMI vier Compute-Knotengruppen verwende?

Sie müssen die AWS PCS Software auf dieselbe Version aktualisieren, um die neue Slurm-Funktionalität nutzen zu können. Für eine vollständige AWS PCS Unterstützung müssen alle Slurm-Komponenten unterstützte Versionen verwenden. Zusammenfassend:

- Wir sind in der Lage, vollen Support zu bieten, wenn der Cluster-Controller und alle Komponenten (AWS PCSPakete) in Ihren AWS-Konto beiden die unterstützten Versionen verwenden.
- AWS PCSist so konzipiert, dass ein Cluster gestoppt wird, wenn die Slurm-Version seines Controllers erreicht wirdEOSL.
- Wenn die Slurm-Version von Komponenten in Ihrer AWS-Konto Reichweite istEOSL, wird Ihr Cluster nicht unterstützt.

In welcher Reihenfolge sollte ich die Komponenten in meinem Cluster aktualisieren?

Sie müssen die Slurm-Version Ihres Cluster-Controllers aktualisieren, bevor Sie eine AMI mit einer neueren Slurm-Version verwenden. Sie aktualisieren eine Compute-Knotengruppe, um die zu verwenden. AMI AWS PCSverwendet dieAMI, um neue EC2 Instanzen in der Compute-Knotengruppe zu starten. AWS PCSaktualisiert keine vorhandenen EC2 Instanzen mit laufenden Jobs; AWS PCS ist so konzipiert, dass diese Instanzen beendet werden, nachdem ihre Jobs abgeschlossen sind.

AWS PCSBietet erweiterten Support für Slurm-Versionen?

Nein. Wir werden Ihnen detaillierte Informationen über erweiterte Support-Optionen, einschließlich aller zusätzlichen Kosten und der spezifischen Support-Abdeckung, mitteilen.

Sicherheit im AWS Parallel-Computing-Dienst

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 AWS Verantwortung von Ihnen und Ihnen. Das Modell der geteilten Verantwortung beschreibt dies als Sicherheit der Cloud selbst 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. Externe 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 Nutzung anwenden können AWS PCS. In den folgenden Themen erfahren Sie, wie Sie die Konfiguration vornehmen AWS PCS, um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie erfahren auch, wie Sie andere AWS Dienste nutzen können, die Sie bei der Überwachung und Sicherung Ihrer AWS PCS Ressourcen unterstützen.

Themen

- Serviceübergreifende Confused-Deputy-Prävention
- Bewährte Sicherheitsmethoden für AWS Parallel Computing Service

Datenschutz im AWS Parallel Computing Service

Das Modell der AWS gemeinsamen Verantwortung und geteilter Verantwortung gilt für den Datenschutz im 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 im Abschnitt Datenschutz FAQ. Informationen zum Datenschutz in Europa finden Sie im AWS Shared Responsibility Model und im GDPR Blogbeitrag auf dem AWS Security Blog.

Aus Datenschutzgründen empfehlen wir, dass Sie Ihre 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 eine Multi-Faktor-Authentifizierung (MFA).
- Verwenden SieSSL/TLS, um mit AWS Ressourcen zu kommunizieren. Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Einrichtung API und Protokollierung von Benutzeraktivitäten mit AWS CloudTrail.
- 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 FIPS 140-3 validierte kryptografische Module für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine benötigenAPI, verwenden Sie einen Endpunkt. FIPS Weitere Informationen zu den verfügbaren FIPS Endpunkten finden Sie unter Federal Information Processing Standard () 140-3. FIPS

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 der Konsole arbeiten AWS PCS oder sie anderweitig AWS-Services verwenden,API, AWS

CLI oder. AWS SDKs Alle Daten, die Sie in Tags oder Freitextfelder eingeben, die für Namen verwendet werden, können für Abrechnungs- oder Diagnoseprotokolle verwendet werden. Wenn Sie einem externen Server eine URL zur Verfügung stellen, empfehlen wir dringend, dass Sie keine Anmeldeinformationen in den angebenURL, um Ihre Anfrage an diesen Server zu überprüfen.

Verschlüsselung im Ruhezustand

Die Verschlüsselung ist standardmäßig für ruhende Daten aktiviert, wenn Sie einen AWS Parallel Computing Service (AWS PCS) -Cluster mit dem AWS Management Console, AWS CLI AWS PCSAPI, oder erstellen AWS SDKs. AWS PCSverwendet einen AWS eigenen KMS Schlüssel, um Daten im Ruhezustand zu verschlüsseln. Weitere Informationen finden Sie unter Kundenschlüssel und AWS Schlüssel im AWS KMS Entwicklerhandbuch. Das Clustergeheimnis wird im verwalteten KMS Schlüssel von Secrets Manager 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 PCSlöscht 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 werden Daten und Metadaten AWS PCS automatisch verschlüsselt, bevor sie in das Dateisystem geschrieben werden. Das verschlüsselte Dateisystem verwendet den Industriestandard-Verschlüsselungsalgorithmus AES -256 für Daten im Ruhezustand.

Verschlüsselung während der Übertragung

Ihre Verbindungen AWS PCS API verwenden TLS Verschlüsselung mit dem Signaturprozess von Signature Version 4, unabhängig davon, ob Sie AWS Command Line Interface (AWS CLI) oder verwenden. AWS SDKs Weitere Informationen finden Sie im AWS Identity and Access Management Benutzerhandbuch unter Signieren von AWS API Anfragen. AWS verwaltet die Zugriffskontrolle API anhand der IAM Richtlinien für die Sicherheitsanmeldedaten, die Sie für die Verbindung verwenden.

AWS PCSverwendetTLS, 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 zwischen Cluster-Instanzen fließen, bleiben lokal auf der Ebene EC2 VPC und unterliegen daher der Verschlüsselung, wenn diese Instanzen die VPC 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 PCSverwendet einen AWS eigenen KMS Schlüssel zum Verschlüsseln von Daten. Weitere Informationen finden Sie unter Kundenschlüssel und AWS Schlüssel im AWS KMS Entwicklerhandbuch. Das Clustergeheimnis wird im verwalteten KMS Schlüssel von Secrets Manager gespeichert AWS Secrets Manager und mit diesem verschlüsselt. Weitere Informationen finden Sie unter Arbeiten mit Clustergeheimnissen in AWS PCS.

Datenschutz für den Datenverkehr zwischen Netzwerken

AWS PCSDie Rechenressourcen für einen Cluster befinden sich innerhalb von 1 VPC im Konto des Kunden. Daher verbleibt der gesamte interne AWS PCS Dienstverkehr 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, unseren Systems Manager zu verwendenSSH, 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 den AWS PCS API zu, um administrative Aufgaben für den Dienst auszuführen. Sie und Ihre Benutzer greifen auf die Slurm-Endpunktports zu, um direkt mit dem Scheduler zu interagieren.

Datenverkehr verschlüsseln API

Um auf die zugreifen zu können AWS PCSAPI, müssen die Clients Transport Layer Security (TLS) 1.2 oder höher unterstützen. Wir benötigen TLS 1.2 und empfehlen TLS 1.3. Kunden müssen auch Cipher Suites mit Perfect Forward Secrecy (PFS) unterstützen, wie Ephemeral Diffie-Hellman () oder Elliptic Curve Diffie-Hellman Ephemeral (DHE). ECDHE Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi. Darüber hinaus müssen Anfragen mithilfe einer Zugriffsschlüssel-ID und eines geheimen Zugriffsschlüssels, der einem Prinzipal zugeordnet ist, signiert werden. IAM Sie können AWS Security Token Service (AWS STS) auch verwenden, um temporäre Sicherheitsanmeldeinformationen zum Signieren von Anfragen zu generieren.

Den Datenverkehr verschlüsseln

Die Verschlüsselung von Daten während der Übertragung wird von unterstützten EC2 Instanzen aus aktiviert, die auf den Scheduler-Endpunkt zugreifen, und zwischen ComputeNodeGroup Instanzen innerhalb von. AWS Cloud Weitere Informationen finden Sie unter Verschlüsselung während der Übertragung.

Greifen Sie über einen Schnittstellenendpunkt ()AWS PrivateLink auf den AWS Parallel Computing Service zu

Sie können AWS PrivateLink es verwenden, um eine private Verbindung zwischen Ihnen VPC und dem AWS Parallel Computing Service (AWS PCS) herzustellen. Sie können darauf zugreifen, AWS PCS als ob es in Ihrem wäreVPC, ohne ein Internet-Gateway, ein NAT Gerät, eine Verbindung oder AWS Direct Connect eine VPN Verbindung verwenden zu müssen. Für den Zugriff auf Ihre Instanzen sind VPC keine öffentlichen IP-Adressen erforderlich AWS PCS.

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. Dabei handelt es sich um vom Anforderer verwaltete Netzwerkschnittstellen, die als Einstiegspunkt für den Datenverkehr dienen, für den. AWS PCS

Weitere Informationen finden Sie AWS PrivateLink im Handbuch unter Access AWS-Services through.AWS PrivateLink

Überlegungen zu AWS PCS

Bevor Sie einen Schnittstellenendpunkt für einrichten AWS PCS, lesen Sie den Artikel Zugriff auf einen AWS Dienst mithilfe eines VPC Schnittstellenendpunkts im AWS PrivateLink Handbuch.

AWS PCSunterstützt das Aufrufen all seiner API Aktionen über den Schnittstellenendpunkt.

Wenn Sie VPC keinen direkten Internetzugang haben, müssen Sie einen VPC Endpunkt konfigurieren, damit Ihre Compute-Knotengruppen-Instances die AWS PCS `RegisterComputeNodeGroupInstance`APIAktion aufrufen können.

Erstellen Sie einen Schnittstellen-Endpunkt für AWS PCS

Sie können einen Schnittstellenendpunkt für die AWS PCS Verwendung entweder der VPC Amazon-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
```

Ersetzen *`region`* mit der ID des, in AWS-Region dem der Endpunkt erstellt werden soll, z. `us-east-1` B.

Wenn Sie Private DNS für den Schnittstellenendpunkt aktivieren, können Sie API Anfragen an die AWS PCS Verwendung DNS des regionalen Standardnamens 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 Schnittstellenendpunkt anhängen können. Die standardmäßige Endpunktrichtlinie ermöglicht den vollen Zugriff AWS PCS über den Schnittstellenendpunkt. Um den Zugriff AWS PCS von Ihrem aus zu kontrollierenVPC, 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 Aktionen AWS PCS

Im Folgenden finden Sie ein Beispiel für eine benutzerdefinierte Endpunktrichtlinie. Wenn Sie diese Richtlinie an Ihren Schnittstellenendpunkt anhängen, gewährt sie allen Prinzipalen des Clusters Zugriff auf die aufgelisteten AWS PCS Aktionen mit den angegebenen *cluster-id*. Ersetzen *region* durch die ID AWS-Region des Clusters, z. us-east-1 B. Ersetzen *account-id* mit der 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 Parallel Computing Service

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service , den Zugriff auf AWS Ressourcen sicher zu kontrollieren. IAMAdministratoren kontrollieren, wer authentifiziert

(angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um AWS PCS Ressourcen zu verwenden. IAMist eine AWS-Service , die Sie ohne zusätzliche Kosten verwenden können.

Themen

Zielgruppe

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, in der Sie arbeiten AWS PCS.

Dienstbenutzer — Wenn Sie den AWS PCS Dienst für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die erforderlichen Anmeldeinformationen und Berechtigungen zur Verfügung. Wenn Sie für Ihre Arbeit mehr AWS PCS Funktionen verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Fuktionsweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anzufordern müssen. Wenn Sie in nicht auf eine Funktion zugreifen können AWS PCS, finden Sie weitere Informationen unterProblembehandlung bei Identität und Zugriff auf den AWS Parallel Computing Service.

Serviceadministrator — Wenn Sie in Ihrem Unternehmen für die AWS PCS Ressourcen verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf AWS PCS. Es ist Ihre Aufgabe, zu bestimmen, auf welche AWS PCS Funktionen und Ressourcen Ihre Servicebenutzer zugreifen sollen. Anschließend müssen Sie Anfragen an Ihren IAM Administrator senden, um die Berechtigungen Ihrer Servicebenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die grundlegenden

Konzepte von zu verstehenIAM. Weitere Informationen darüber, wie Ihr Unternehmen IAM mit verwenden kann AWS PCS, finden Sie unterSo funktioniert AWS Parallel Computing Service mit IAM.

IAMAdministrator — Wenn Sie ein IAM Administrator sind, möchten Sie vielleicht mehr darüber erfahren, wie Sie Richtlinien schreiben können, um den Zugriff darauf zu verwalten AWS PCS. Beispiele für AWS PCS identitätsbasierte Richtlinien, die Sie in verwenden könnenIAM, finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Parallel Computing Service AWS

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen sich als IAM Benutzer authentifizieren (angemeldet bei AWS) oder indem Sie eine IAM Rolle übernehmen. Root-Benutzer des AWS-Kontos

Sie können sich AWS als föderierte Identität anmelden, indem Sie Anmeldeinformationen verwenden, die über eine Identitätsquelle bereitgestellt wurden. AWS IAM Identity Center (IAMIdentity Center-) Nutzer, die Single-Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie sich als föderierte Identität anmelden, hat Ihr Administrator zuvor einen Identitätsverbund mithilfe von Rollen eingerichtet. IAM Wenn Sie AWS mithilfe eines Verbunds darauf zugreifen, übernehmen Sie indirekt eine Rolle.

Je nachdem, welcher Benutzertyp Sie sind, können Sie sich beim AWS Management Console oder beim AWS Zugangsportal anmelden. Weitere Informationen zur Anmeldung finden Sie AWS unter So melden Sie sich bei Ihrem an AWS-Konto im AWS-Anmeldung Benutzerhandbuch.

Wenn Sie AWS programmgesteuert darauf zugreifen, AWS stellt es ein Software Development Kit (SDK) und eine Befehlszeilenschnittstelle (CLI) bereit, mit der Sie Ihre Anfragen mithilfe Ihrer Anmeldeinformationen kryptografisch signieren können. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode, um Anfragen selbst zu signieren, finden Sie im IAMBenutzerhandbuch unter AWS API Anfragen signieren.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen angeben. AWS Empfiehlt beispielsweise, die Multi-Faktor-Authentifizierung (MFA) zu verwenden, um die Sicherheit Ihres Kontos zu erhöhen. Weitere Informationen finden Sie unter Multi-Faktor-Authentifizierung im AWS IAM Identity Center Benutzerhandbuch und Verwenden der Multi-Faktor-Authentifizierung (MFA) AWS im IAM Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle AWS-Services Ressourcen im Konto hat. Diese Identität wird als AWS-Konto Root-Benutzer bezeichnet. Sie können darauf zugreifen, indem Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, mit denen Sie das Konto erstellt haben. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen und verwenden Sie diese, um die Aufgaben auszuführen, die nur der Root-Benutzer ausführen kann. Eine vollständige Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie im Benutzerhandbuch unter Aufgaben, für die Root-Benutzeranmeldedaten erforderlich sind. IAM

Verbundidentität

Als bewährte Methode sollten menschliche Benutzer, einschließlich Benutzer, die Administratorzugriff benötigen, für den Zugriff AWS-Services mithilfe temporärer Anmeldeinformationen den Verbund mit einem Identitätsanbieter verwenden.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensbenutzerverzeichnis, einem Web-Identitätsanbieter AWS Directory Service, dem Identity Center-Verzeichnis oder einem beliebigen Benutzer, der mithilfe AWS-Services von Anmeldeinformationen zugreift, die über eine Identitätsquelle bereitgestellt wurden. Wenn föderierte Identitäten darauf zugreifen AWS-Konten, übernehmen sie Rollen, und die Rollen stellen temporäre Anmeldeinformationen bereit.

Für die zentrale Zugriffsverwaltung empfehlen wir Ihnen, AWS IAM Identity Center zu verwenden. Sie können Benutzer und Gruppen in IAM Identity Center erstellen, oder Sie können eine Verbindung zu einer Gruppe von Benutzern und Gruppen in Ihrer eigenen Identitätsquelle herstellen und diese synchronisieren, um sie in all Ihren AWS-Konten Anwendungen zu verwenden. Informationen zu IAM Identity Center finden Sie unter Was ist IAM Identity Center? im AWS IAM Identity Center Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein IAMBenutzer ist eine Identität innerhalb Ihres Unternehmens AWS-Konto , die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. Wir empfehlen, sich nach Möglichkeit auf temporäre Anmeldeinformationen zu verlassen, anstatt IAM Benutzer mit langfristigen Anmeldeinformationen wie Passwörtern und Zugriffsschlüsseln zu erstellen. Wenn Sie jedoch spezielle Anwendungsfälle haben, für die langfristige Anmeldeinformationen von IAM Benutzern erforderlich sind, empfehlen wir, die Zugriffsschlüssel abwechselnd zu verwenden. Weitere

Informationen finden Sie im Benutzerhandbuch unter Regelmäßiges Rotieren von Zugriffsschlüsseln für Anwendungsfälle, für die IAM langfristige Anmeldeinformationen erforderlich sind.

Eine IAMGruppe ist eine Identität, die eine Sammlung von IAM Benutzern angibt. Sie können sich nicht als Gruppe anmelden. Mithilfe von Gruppen können Sie Berechtigungen für mehrere Benutzer gleichzeitig angeben. Gruppen vereinfachen die Verwaltung von Berechtigungen, wenn es zahlreiche Benutzer gibt. Sie könnten beispielsweise eine Gruppe benennen IAMAdminsund dieser Gruppe Berechtigungen zur Verwaltung von IAM Ressourcen erteilen.

Benutzer unterscheiden sich von Rollen. Ein Benutzer ist einer einzigen Person oder Anwendung eindeutig zugeordnet. Eine Rolle kann von allen Personen angenommen werden, die sie benötigen. Benutzer besitzen dauerhafte Anmeldeinformationen. Rollen stellen temporäre Anmeldeinformationen bereit. Weitere Informationen finden Sie unter Wann sollte ein IAM Benutzer (statt einer Rolle) erstellt werden? im IAMBenutzerhandbuch.

IAMRollen

Eine IAMRolle ist eine Identität innerhalb von Ihnen AWS-Konto , für die bestimmte Berechtigungen gelten. Sie ähnelt einem IAM Benutzer, ist jedoch keiner bestimmten Person zugeordnet. Sie können vorübergehend eine IAM Rolle in der übernehmen, AWS Management Console indem Sie die Rollen wechseln. Sie können eine Rolle übernehmen, indem Sie eine AWS CLI AWS API OR-Operation aufrufen oder eine benutzerdefinierte Operation verwendenURL. Weitere Informationen zu Methoden zur Verwendung von Rollen finden Sie unter Verwenden von IAM Rollen im IAMBenutzerhandbuch.

IAMRollen mit temporären Anmeldeinformationen sind in den folgenden Situationen nützlich:

- Verbundbenutzerzugriff – Um einer Verbundidentität Berechtigungen zuzuweisen, erstellen Sie eine Rolle und definieren Berechtigungen für die Rolle. Wird eine Verbundidentität authentifiziert, so wird die Identität der Rolle zugeordnet und erhält die von der Rolle definierten Berechtigungen. Informationen zu Rollen für den Verbund finden Sie im IAMBenutzerhandbuch unter Erstellen einer Rolle für einen externen Identitätsanbieter. Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Um zu kontrollieren, worauf Ihre Identitäten nach der Authentifizierung zugreifen können, korreliert IAM Identity Center den Berechtigungssatz mit einer Rolle in. IAM Informationen zu Berechtigungssätzen finden Sie unter Berechtigungssätze im AWS IAM Identity Center -Benutzerhandbuch.
- Temporäre IAM Benutzerberechtigungen — Ein IAM Benutzer oder eine Rolle kann eine IAM Rolle übernehmen, um vorübergehend verschiedene Berechtigungen für eine bestimmte Aufgabe zu übernehmen.

- Kontoübergreifender Zugriff — Sie können eine IAM Rolle verwenden, um einer Person (einem vertrauenswürdigen Principal) in einem anderen Konto den Zugriff auf Ressourcen in Ihrem Konto zu ermöglichen. Rollen stellen die primäre Möglichkeit dar, um kontoübergreifendem Zugriff zu gewähren. Bei einigen können Sie AWS-Services jedoch eine Richtlinie direkt an eine Ressource anhängen (anstatt eine Rolle als Proxy zu verwenden). Informationen zum Unterschied zwischen Rollen und ressourcenbasierten Richtlinien für den kontenübergreifenden Zugriff finden Sie IAMim Benutzerhandbuch unter Kontoübergreifender Ressourcenzugriff. IAM
- Serviceübergreifender Zugriff — Einige AWS-Services verwenden Funktionen in anderen. AWS-Services Wenn Sie beispielsweise in einem Service einen Anruf tätigen, ist es üblich, dass dieser Service Anwendungen in Amazon ausführt EC2 oder Objekte in Amazon S3 speichert. Ein Dienst kann dies mit den Berechtigungen des aufrufenden Prinzipals mit einer Servicerolle oder mit einer serviceverknüpften Rolle tun.
  - Zugriffssitzungen weiterleiten (FAS) — Wenn Sie einen IAM Benutzer oder eine Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FASverwendet die Berechtigungen des Prinzipals, der an aufruft AWS-Service, kombiniert mit der Anforderung, Anfragen AWS-Service an nachgelagerte Dienste zu stellen. FASAnfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien beim Stellen von FAS Anfragen finden Sie unter Zugriffssitzungen weiterleiten.
  - Servicerolle — Eine Servicerolle ist eine IAMRolle, die ein Dienst übernimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM Administrator kann eine Servicerolle von innen heraus erstellen, ändern und löschenIAM. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Erstellen einer Rolle zum Delegieren von Berechtigungen AWS-Service an eine.
  - Dienstbezogene Rolle — Eine dienstverknüpfte Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Servicebezogene Rollen erscheinen in Ihrem Dienst AWS-Konto und gehören dem Dienst. Ein IAM Administrator kann die Berechtigungen für dienstbezogene Rollen anzeigen, aber nicht bearbeiten.
- Auf Amazon ausgeführte Anwendungen EC2 — Sie können eine IAM Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2 Instance ausgeführt werden und AWS API Anfragen stellen AWS CLI . Dies ist dem Speichern von Zugriffsschlüsseln innerhalb der EC2 Instance vorzuziehen. Um einer EC2 Instanz eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein

Instanzprofil, das an die Instanz angehängt ist. Ein Instanzprofil enthält die Rolle und ermöglicht Programmen, die auf der EC2 Instanz ausgeführt werden, temporäre Anmeldeinformationen abzurufen. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Verwenden einer IAM Rolle zur Erteilung von Berechtigungen für Anwendungen, die auf EC2 Amazon-Instances ausgeführt werden.

Informationen darüber, ob Sie IAM Rollen oder IAM Benutzer verwenden sollten, finden Sie im Benutzerhandbuch unter Wann sollte eine IAM Rolle (anstelle eines IAM Benutzers) erstellt werden.

Verwalten des Zugriffs mit Richtlinien

Sie steuern den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie ist ein Objekt, AWS das, wenn es einer Identität oder Ressource zugeordnet ist, deren Berechtigungen definiert. AWS wertet diese Richtlinien aus, wenn ein Prinzipal (Benutzer, Root-Benutzer oder Rollensitzung) eine Anfrage stellt. Berechtigungen in den Richtlinien bestimmen, ob die Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden in AWS Form von JSON Dokumenten gespeichert. Weitere Informationen zur Struktur und zum Inhalt von JSON Richtliniendokumenten finden Sie im IAMBenutzerhandbuch unter Überblick über JSON Richtlinien.

Administratoren können mithilfe von AWS JSON Richtlinien festlegen, wer Zugriff auf was hat. Das bedeutet, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Um Benutzern die Erlaubnis zu erteilen, Aktionen mit den Ressourcen durchzuführen, die sie benötigen, kann ein IAM Administrator IAM Richtlinien erstellen. Der Administrator kann dann die IAM Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen übernehmen.

IAMRichtlinien definieren Berechtigungen für eine Aktion, unabhängig von der Methode, mit der Sie den Vorgang ausführen. Angenommen, es gibt eine Richtlinie, die Berechtigungen für die `iam:GetRole`-Aktion erteilt. Ein Benutzer mit dieser Richtlinie kann Rolleninformationen aus dem AWS Management Console AWS CLI, dem oder dem abrufen AWS API.

Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind Dokumente mit JSON Berechtigungsrichtlinien, 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 einer identitätsbasierten Richtlinie finden Sie unter IAMRichtlinien erstellen im Benutzerhandbuch. IAM

Identitätsbasierte Richtlinien können weiter als Inline-Richtlinien oder verwaltete Richtlinien kategorisiert werden. Inline-Richtlinien sind direkt in einen einzelnen Benutzer, eine einzelne Gruppe oder eine einzelne Rolle eingebettet. Verwaltete Richtlinien sind eigenständige Richtlinien, die Sie mehreren Benutzern, Gruppen und Rollen in Ihrem System zuordnen können. AWS-Konto Zu den verwalteten Richtlinien gehören AWS verwaltete Richtlinien und vom Kunden verwaltete Richtlinien. Informationen dazu, wie Sie zwischen einer verwalteten Richtlinie oder einer Inline-Richtlinie wählen können, finden Sie im IAMBenutzerhandbuch unter Auswahl zwischen verwalteten Richtlinien und Inline-Richtlinien.

Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON Richtliniendokumente, die Sie an eine Ressource anhängen. Beispiele für ressourcenbasierte Richtlinien sind IAM Rollenvertrauensrichtlinien und Amazon S3 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

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien nicht IAM in einer ressourcenbasierten Richtlinie verwenden.

Zugriffskontrolllisten () ACLs

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLsähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON Richtliniendokumentformat.

Amazon S3 und AWS WAF Amazon VPC sind Beispiele für Dienste, die Unterstützung bietenACLs. Weitere Informationen finden Sie unter Übersicht über ACLs die Zugriffskontrollliste (ACL) im Amazon Simple Storage Service Developer Guide.

Weitere Richtlinientypen

AWS unterstützt zusätzliche, weniger verbreitete Richtlinientypen. Diese Richtlinientypen können die maximalen Berechtigungen festlegen, die Ihnen von den häufiger verwendeten Richtlinientypen erteilt werden können.

- Berechtigungsgrenzen — Eine Berechtigungsgrenze ist eine erweiterte Funktion, mit der Sie die maximalen Berechtigungen festlegen, die eine identitätsbasierte Richtlinie einer IAM Entität (IAMBenutzer oder Rolle) gewähren kann. Sie können eine Berechtigungsgrenze für eine Entität festlegen. Die daraus resultierenden Berechtigungen sind der Schnittpunkt der identitätsbasierten Richtlinien einer Entität und ihrer Berechtigungsgrenzen. Ressourcenbasierte Richtlinien, die den Benutzer oder die Rolle im Feld `Principal` angeben, werden nicht durch Berechtigungsgrenzen eingeschränkt. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen zu Berechtigungsgrenzen finden Sie im IAMBenutzerhandbuch unter Berechtigungsgrenzen für IAM Entitäten.
- Dienststeuerungsrichtlinien (SCPs) — SCPs sind JSON Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OU) in festlegen AWS Organizations. AWS Organizations ist ein Dienst zur Gruppierung und zentralen Verwaltung mehrerer Geräte AWS-Konten , die Ihrem Unternehmen gehören. Wenn Sie alle Funktionen in einer Organisation aktivieren, können Sie Richtlinien zur Servicesteuerung (SCPs) auf einige oder alle Ihre Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Entitäten Root-Benutzer des AWS-Kontos. Weitere Informationen zu Organizations und SCPs finden Sie unter Richtlinien zur Servicesteuerung im AWS Organizations Benutzerhandbuch.
- Sitzungsrichtlinien – Sitzungsrichtlinien sind erweiterte Richtlinien, die Sie als Parameter übergeben, wenn Sie eine temporäre Sitzung für eine Rolle oder einen verbundenen Benutzer programmgesteuert erstellen. Die resultierenden Sitzungsberechtigungen sind eine Schnittmenge der auf der Identität des Benutzers oder der Rolle basierenden Richtlinien und der Sitzungsrichtlinien. Berechtigungen können auch aus einer ressourcenbasierten Richtlinie stammen. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Sitzungsrichtlinien.

Mehrere Richtlinientypen

Wenn mehrere auf eine Anforderung mehrere Richtlinientypen angewendet werden können, sind die entsprechenden Berechtigungen komplizierter. Informationen darüber, wie AWS bestimmt

wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie im IAMBenutzerhandbuch unter Bewertungslogik für Richtlinien.

So funktioniert AWS Parallel Computing Service mit IAM

Informieren Sie sich vor der Verwendung IAM zur Verwaltung des Zugriffs auf AWS PCS, welche IAM Funktionen zur Verwendung verfügbar sind AWS PCS.

IAMFunktionen, die Sie mit AWS Parallel Computing Service verwenden können

IAMFunktion	AWS PCSUnterstützung
Identitätsbasierte Richtlinien	Ja
Ressourcenbasierte Richtlinien	Nein
Richtlinienaktionen	Ja
Richtlinienressourcen	Ja
Richtlinienbedingungsschlüssel (services pezifisch)	Ja
ACLs	Nein
ABAC(Tags in Richtlinien)	Ja
Temporäre Anmeldeinformationen	Ja
Hauptberechtigungen	Ja
Servicerollen	Nein
Serviceverknüpfte Rollen	Ja

Einen allgemeinen Überblick darüber, wie AWS PCS und wie andere AWS Dienste mit den meisten IAM Funktionen funktionieren, finden Sie IAM im IAMBenutzerhandbuch unter AWS Dienste, die mit funktionieren.

Identitätsbasierte Richtlinien für AWS PCS

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identitätsbasierte Richtlinien sind Dokumente mit JSON Berechtigungsrichtlinien, 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 einer identitätsbasierten Richtlinie finden Sie unter IAMRichtlinien erstellen im Benutzerhandbuch. IAM

Mit IAM identitätsbasierten Richtlinien können Sie zulässige oder verweigerte Aktionen und Ressourcen sowie die Bedingungen angeben, unter denen Aktionen zulässig oder verweigert werden. Sie können den Prinzipal nicht in einer identitätsbasierten Richtlinie angeben, da er für den Benutzer oder die Rolle gilt, dem er zugeordnet ist. Weitere Informationen zu allen Elementen, die Sie in einer JSON Richtlinie verwenden können, finden Sie im IAMBenutzerhandbuch unter Referenz zu IAM JSON Richtlinienelementen.

Beispiele für identitätsbasierte Richtlinien für AWS PCS

Beispiele für AWS PCS identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Parallel Computing Service AWS

Ressourcenbasierte Richtlinien finden Sie in AWS PCS

Unterstützt ressourcenbasierte Richtlinien: Nein

Ressourcenbasierte Richtlinien sind JSON Richtliniendokumente, die Sie an eine Ressource anhängen. Beispiele für ressourcenbasierte Richtlinien sind IAM Rollenvertrauensrichtlinien und Amazon S3 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 den kontoübergreifenden Zugriff zu ermöglichen, können Sie in einer ressourcenbasierten Richtlinie ein ganzes Konto oder IAM Entitäten in einem anderen Konto als Prinzipal angeben. Durch das Hinzufügen eines kontoübergreifenden Auftraggebers zu einer ressourcenbasierten Richtlinie ist nur die halbe Vertrauensbeziehung eingerichtet. Wenn sich der Prinzipal und die Ressource

unterscheiden AWS-Konten, muss ein IAM Administrator des vertrauenswürdigen Kontos auch der Prinzipalentität (Benutzer oder Rolle) die Berechtigung zum Zugriff auf die Ressource gewähren. Sie erteilen Berechtigungen, indem Sie der juristischen Stelle eine identitätsbasierte Richtlinie anfügen. Wenn jedoch eine ressourcenbasierte Richtlinie Zugriff auf einen Prinzipal in demselben Konto gewährt, ist keine zusätzliche identitätsbasierte Richtlinie erforderlich. Weitere Informationen finden Sie IAMim IAMBenutzerhandbuch unter Kontenübergreifender Ressourcenzugriff.

Politische Maßnahmen für AWS PCS

Unterstützt Richtlinienaktionen: Ja

Administratoren können mithilfe von AWS JSON Richtlinien angeben, wer Zugriff auf was hat. Das bedeutet, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das `Action` Element einer JSON Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Richtlinienaktionen haben normalerweise denselben Namen wie der zugehörige AWS API Vorgang. Es gibt einige Ausnahmen, z. B. Aktionen, für die nur eine Genehmigung erforderlich ist und für die es keinen entsprechenden Vorgang gibt. API Es gibt auch einige Operationen, die mehrere Aktionen in einer Richtlinie erfordern. Diese zusätzlichen Aktionen werden als abhängige Aktionen bezeichnet.

Schließen Sie Aktionen in eine Richtlinie ein, um Berechtigungen zur Durchführung der zugeordneten Operation zu erteilen.

Eine Liste der AWS PCS Aktionen finden Sie unter Von AWS Parallel Computing Service definierte Aktionen in der Serviceautorisierungsreferenz.

Bei Richtlinienaktionen wird vor der Aktion das folgende Präfix AWS PCS verwendet:

```
pcs
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Kommata:

```
"Action": [
      "pcs:action1",
      "pcs:action2"
         ]
```

Beispiele für AWS PCS identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Parallel Computing Service AWS

Politische Ressourcen für AWS PCS

Unterstützt Richtlinienressourcen: Ja

Administratoren können mithilfe von AWS JSON Richtlinien festlegen, wer Zugriff auf was hat. Das bedeutet, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das `Resource` JSON Richtlinienelement gibt das Objekt oder die Objekte an, für die die Aktion gilt. Anweisungen müssen entweder ein – `Resource`oder ein `NotResource`-Element enthalten. Es hat sich bewährt, eine Ressource mit ihrem Amazon-Ressourcennamen (ARN) anzugeben. Sie können dies für Aktionen tun, die einen bestimmten Ressourcentyp unterstützen, der als Berechtigungen auf Ressourcenebene bezeichnet wird.

Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, z. B. Auflistungsoperationen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

```
"Resource": "*"
```

Eine Liste der AWS PCS Ressourcentypen und ihrer ARNs Eigenschaften finden Sie unter Von AWS Parallel Computing Service definierte Ressourcen in der Service Authorization Reference. Informationen darüber, mit welchen Aktionen Sie die ARN einzelnen Ressourcen spezifizieren können, finden Sie unter Von AWS Parallel Computing Service definierte Aktionen.

Beispiele für AWS PCS identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Parallel Computing Service AWS

Bedingungsschlüssel für Richtlinien für AWS PCS

Unterstützt servicespezifische Richtlinienbedingungsschlüssel: Ja

Administratoren können mithilfe von AWS JSON Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das Element `Condition` (oder `Condition` block) ermöglicht Ihnen die Angabe der Bedingungen, unter denen eine Anweisung wirksam ist. Das Element `Condition` ist optional. 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.

Wenn Sie mehrere `Condition`-Elemente in einer Anweisung oder mehrere Schlüssel in einem einzelnen `Condition`-Element angeben, wertet AWS diese mittels einer logischen `AND`-Operation aus. Wenn Sie mehrere Werte für einen einzelnen Bedingungsschlüssel angeben, AWS wertet die Bedingung mithilfe einer logischen `OR` Operation aus. Alle Bedingungen müssen erfüllt werden, bevor die Berechtigungen der Anweisung gewährt werden.

Sie können auch Platzhaltervariablen verwenden, wenn Sie Bedingungen angeben. Sie können einem IAM Benutzer beispielsweise nur dann Zugriff auf eine Ressource gewähren, wenn sie mit seinem IAM Benutzernamen gekennzeichnet ist. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter IAMRichtlinienelemente: Variablen und Tags.

AWS unterstützt globale Bedingungsschlüssel und dienstspezifische Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontext-Schlüssel für AWS globale Bedingungen im IAMBenutzerhandbuch.

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 AWS PCS identitätsbasierte Richtlinien finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Parallel Computing Service AWS

ACLsin AWS PCS

UnterstütztACLs: Nein

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLsähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON Richtliniendokumentformat.

ABACmit AWS PCS

Unterstützt ABAC (Tags in Richtlinien): Ja

Die attributbasierte Zugriffskontrolle (ABAC) ist eine Autorisierungsstrategie, die Berechtigungen auf der Grundlage von Attributen definiert. In werden AWS diese Attribute als Tags bezeichnet. Sie können Tags an IAM Entitäten (Benutzer oder Rollen) und an viele AWS Ressourcen anhängen. Das Markieren von Entitäten und Ressourcen ist der erste Schritt vonABAC. Anschließend entwerfen Sie ABAC Richtlinien, die Operationen zulassen, wenn das Tag des Prinzipals mit dem Tag auf der Ressource übereinstimmt, auf die er zugreifen möchte.

ABACist hilfreich in Umgebungen, die schnell wachsen, und hilft in Situationen, in denen die Richtlinienverwaltung umständlich wird.

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 finden Sie ABAC unter Was ist? ABAC im IAMBenutzerhandbuch. Ein Tutorial mit Schritten zur Einrichtung finden Sie im ABAC Benutzerhandbuch unter Verwenden der attributebasierten Zugriffskontrolle (ABAC). IAM

Verwenden temporärer Anmeldeinformationen mit AWS PCS

Unterstützt temporäre Anmeldeinformationen: Ja

Einige funktionieren AWS-Services nicht, wenn Sie sich mit temporären Anmeldeinformationen anmelden. Weitere Informationen, einschließlich Informationen darüber, AWS-Services wie Sie mit temporären Anmeldeinformationen arbeiten können AWS-Services , finden Sie IAM im IAMBenutzerhandbuch unter Informationen zum Arbeiten mit.

Sie verwenden temporäre Anmeldeinformationen, wenn Sie sich mit einer anderen AWS Management Console Methode als einem Benutzernamen und einem Kennwort anmelden. Wenn Sie beispielsweise AWS über den Single Sign-On-Link (SSO) Ihres Unternehmens darauf zugreifen, werden bei diesem Vorgang automatisch temporäre Anmeldeinformationen erstellt. Sie erstellen auch automatisch temporäre Anmeldeinformationen, wenn Sie sich als Benutzer bei der Konsole anmelden und dann die Rollen wechseln. Weitere Informationen zum Rollenwechsel finden Sie unter Wechseln zu einer Rolle (Konsole) im IAMBenutzerhandbuch.

Mit dem AWS CLI oder können Sie manuell temporäre Anmeldeinformationen erstellen AWS API. Sie können diese temporären Anmeldeinformationen dann für den Zugriff verwenden AWS. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter Temporäre Sicherheitsanmeldeinformationen unter IAM.

Serviceübergreifende Prinzipalberechtigungen für AWS PCS

Unterstützt Forward-Access-Sitzungen (FAS): Ja

Wenn Sie einen IAM Benutzer oder eine Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FASverwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, kombiniert mit der Anforderung, Anfragen AWS-Service an nachgelagerte Dienste zu stellen. FASAnfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien beim Stellen von FAS Anfragen finden Sie unter Zugriffssitzungen weiterleiten.

Servicerollen für AWS PCS

Unterstützt Servicerollen: Nein

Eine Servicerolle ist eine IAMRolle, die ein Dienst übernimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM Administrator kann eine Servicerolle von innen heraus erstellen, ändern und löschenIAM. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Erstellen einer Rolle zum Delegieren von Berechtigungen AWS-Service an eine.

> **Warning**
>
> Das Ändern der Berechtigungen für eine Servicerolle kann zu AWS PCS Funktionseinschränkungen führen. Bearbeiten Sie Servicerollen nur, AWS PCS wenn Sie dazu eine Anleitung erhalten.

Dienstbezogene Rollen für AWS PCS

Unterstützt dienstbezogene Rollen: Ja

Eine serviceverknüpfte Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Dienstbezogene Rollen werden in Ihrem Dienst angezeigt AWS-Konto und gehören dem Dienst. Ein IAM Administrator kann die Berechtigungen für dienstbezogene Rollen anzeigen, aber nicht bearbeiten.

Einzelheiten zum Erstellen oder Verwalten von dienstbezogenen Rollen finden Sie unter AWS Dienste, die mit funktionieren. IAM Suchen Sie in der Tabelle nach einem Service mit einem `Yes` in der Spalte Service-linked role (Serviceverknüpfte Rolle). Wählen Sie den Link Yes (Ja) aus, um die Dokumentation für die serviceverknüpfte Rolle für diesen Service anzuzeigen.

Beispiele für identitätsbasierte Richtlinien für Parallel Computing Service AWS

Standardmäßig sind Benutzer und Rollen nicht berechtigt, Ressourcen zu erstellen oder zu ändern AWS PCS. Sie können auch keine Aufgaben mithilfe von AWS Management Console, AWS Command Line Interface (AWS CLI) oder ausführen AWS API. Um Benutzern die Berechtigung zu erteilen, Aktionen mit den Ressourcen durchzuführen, die sie benötigen, kann ein IAM Administrator IAM Richtlinien erstellen. Der Administrator kann dann die IAM Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen übernehmen.

Informationen zum Erstellen einer IAM identitätsbasierten Richtlinie anhand dieser JSON Beispieldokumente finden Sie unter IAMRichtlinien erstellen im IAMBenutzerhandbuch.

Weitere Informationen zu Aktionen und Ressourcentypen, die von definiert wurden AWS PCS, 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

- Bewährte Methoden für Richtlinien
- Verwenden der AWS PCS Konsole
- Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

Bewährte Methoden für Richtlinien

Identitätsbasierte Richtlinien legen fest, ob jemand AWS PCS Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder sie löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-

Konto. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen und Empfehlungen:

- Beginnen Sie mit AWS verwalteten Richtlinien und wechseln Sie zu Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um Ihren Benutzern und Workloads zunächst 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 AWS im IAMBenutzerhandbuch unter AWS Verwaltete Richtlinien oder Verwaltete Richtlinien für Jobfunktionen.
- Berechtigungen mit den geringsten Rechten anwenden — Wenn Sie Berechtigungen mit IAM Richtlinien festlegen, gewähren Sie nur die Berechtigungen, die für die Ausführung 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 IAM zum Anwenden von Berechtigungen finden Sie IAMim Benutzerhandbuch unter Richtlinien und Berechtigungen. IAM
- Verwenden Sie Bedingungen in IAM Richtlinien, um den Zugriff weiter einzuschränken — Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen einzuschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben, um anzugeben, dass alle Anfragen mit gesendet werden müssenSSL. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese über einen bestimmten Zweck verwendet werden AWS-Service, z. AWS CloudFormation B. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter IAMJSONRichtlinienelemente: Bedingung.
- Verwenden Sie IAM Access Analyzer, um Ihre IAM Richtlinien zu validieren, um sichere und funktionale Berechtigungen zu gewährleisten. IAM Access Analyzer validiert neue und bestehende Richtlinien, sodass die Richtlinien der IAM Richtliniensprache (JSON) und den IAM bewährten Methoden entsprechen. IAMAccess Analyzer bietet mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen, um Sie bei der Erstellung sicherer und funktionaler Richtlinien zu unterstützen. Weitere Informationen finden Sie unter IAMAccess Analyzer-Richtlinienvalidierung im IAMBenutzerhandbuch.
- Multi-Faktor-Authentifizierung erforderlich (MFA) — Wenn Sie ein Szenario haben, in dem IAM Benutzer oder ein Root-Benutzer erforderlich sind AWS-Konto, aktivieren Sie die Option MFA für zusätzliche Sicherheit. Wenn Sie festlegen möchten, MFA wann API Operationen aufgerufen werden, fügen Sie MFA Bedingungen zu Ihren Richtlinien hinzu. Weitere Informationen finden Sie unter Konfiguration des MFA -geschützten API Zugriffs im IAMBenutzerhandbuch.

Weitere Informationen zu bewährten Methoden finden Sie unter Bewährte Sicherheitsmethoden IAM im IAM Benutzerhandbuch. IAM

Verwenden 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 Anrufe an AWS CLI oder am tätigen, keine Mindestberechtigungen für die Konsole gewähren AWS API. Erlauben Sie stattdessen nur den Zugriff auf die Aktionen, die dem API Vorgang entsprechen, den sie ausführen möchten.

Weitere Informationen zu den Mindestberechtigungen, die für die Verwendung der AWS PCS Konsole erforderlich sind, finden Sie unterMindestberechtigungen für AWS PCS.

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

Dieses Beispiel zeigt, wie Sie eine Richtlinie erstellen könnten, die es IAM Benutzern ermöglicht, die Inline- und verwalteten Richtlinien einzusehen, die mit ihrer Benutzeridentität verknüpft sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe von oder. AWS CLI AWS API

```
{
    "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 Parallel Computing Service

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 allen AWS Kunden zur Verfügung stehen. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie kundenverwaltete 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 im IAMBenutzerhandbuch unter AWS Verwaltete 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 Serviceverknüpfte Rollen für AWS PCS.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `ec2`— Ermöglicht AWS PCS das Erstellen und Verwalten von EC2 Amazon-Ressourcen.
- `iam`— Ermöglicht es AWS PCS, eine servicebezogene Rolle für die EC2 Amazon-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 Cluster-Ressourcen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PermissionsToCreatePCSNetworkInterfaces",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
                "Null": {
                    "aws:RequestTag/AWSPCSManaged": "false"
                }
            }
        },
        {
            "Sid": "PermissionsToCreatePCSNetworkInterfacesInSubnet",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:security-group/*"
            ]
        },
        {
            "Sid": "PermissionsToManagePCSNetworkInterfaces",
            "Effect": "Allow",
            "Action": [
                "ec2:DeleteNetworkInterface",
                "ec2:CreateNetworkInterfacePermission"
            ],
            "Resource": "arn:aws:ec2:*:*:network-interface/*",
            "Condition": {
                "Null": {
```

```
                    "aws:ResourceTag/AWSPCSManaged": "false"
                }
            }
        },
        {
            "Sid": "PermissionsToDescribePCSResources",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeLaunchTemplates",
                "ec2:DescribeLaunchTemplateVersions",
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceTypes",
                "ec2:DescribeInstanceStatus",
                "ec2:DescribeInstanceAttribute",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeKeyPairs",
                "ec2:DescribeImages",
                "ec2:DescribeImageAttribute"
            ],
            "Resource": "*"
        },
        {
            "Sid": "PermissionsToCreatePCSLaunchTemplates",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateLaunchTemplate"
            ],
            "Resource": "arn:aws:ec2:*:*:launch-template/*",
            "Condition": {
                "Null": {
                    "aws:RequestTag/AWSPCSManaged": "false"
                }
            }
        },
        {
            "Sid": "PermissionsToManagePCSLaunchTemplates",
            "Effect": "Allow",
            "Action": [
                "ec2:DeleteLaunchTemplate",
                "ec2:DeleteLaunchTemplateVersions",
                "ec2:CreateLaunchTemplateVersion"
```

```
            ],
            "Resource": "arn:aws:ec2:*:*:launch-template/*",
            "Condition": {
                "Null": {
                    "aws:ResourceTag/AWSPCSManaged": "false"
                }
            }
        },
        {
            "Sid": "PermissionsToTerminatePCSManagedInstances",
            "Effect": "Allow",
            "Action": [
                "ec2:TerminateInstances"
            ],
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "Null": {
                    "aws:ResourceTag/AWSPCSManaged": "false"
                }
            }
        },
        {
            "Sid": "PermissionsToPassRoleToEC2",
            "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": "PermissionsToControlClusterInstanceAttributes",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances",
```

```
                "ec2:CreateFleet"
            ],
            "Resource": [
                "arn:aws:ec2:*::image/*",
                "arn:aws:ec2:*::snapshot/*",
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:network-interface/*",
                "arn:aws:ec2:*:*:security-group/*",
                "arn:aws:ec2:*:*:volume/*",
                "arn:aws:ec2:*:*:key-pair/*",
                "arn:aws:ec2:*:*:launch-template/*",
                "arn:aws:ec2:*:*:placement-group/*",
                "arn:aws:ec2:*:*:capacity-reservation/*",
                "arn:aws:resource-groups:*:*:group/*",
                "arn:aws:ec2:*:*:fleet/*"
            ]
        },
        {
            "Sid": "PermissionsToProvisionClusterInstances",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances",
                "ec2:CreateFleet"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:instance/*"
            ],
            "Condition": {
                "Null": {
                    "aws:RequestTag/AWSPCSManaged": "false"
                }
            }
        },
        {
            "Sid": "PermissionsToTagPCSResources",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringEquals": {
```

```
                    "ec2:CreateAction": [
                        "RunInstances",
                        "CreateLaunchTemplate",
                        "CreateFleet",
                        "CreateNetworkInterface"
                    ]
                }
            }
        },
        {
            "Sid": "PermissionsToPublishMetrics",
            "Effect": "Allow",
            "Action": "cloudwatch:PutMetricData",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "cloudwatch:namespace": "AWS/PCS"
                }
            }
        },
        {
            "Sid": "PermissionsToManageSecret",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:DescribeSecret",
                "secretsmanager:GetSecretValue",
                "secretsmanager:PutSecretValue",
                "secretsmanager:UpdateSecretVersionStage",
                "secretsmanager:DeleteSecret"
            ],
            "Resource": "arn:aws:secretsmanager:*:*:secret:pcs!*",
            "Condition": {
                "StringEquals": {
                    "secretsmanager:ResourceTag/aws:secretsmanager:owningService":
 "pcs",
                    "aws:ResourceAccount": "${aws:PrincipalAccount}"
                }
           }
        }
    ]
}
```

AWS PCSAktualisierungen der AWS verwalteten Richtlinien

Hier finden Sie Informationen zu Aktualisierungen AWS verwalteter Richtlinien, die AWS PCS seit Beginn der Nachverfolgung dieser Änderungen durch diesen Dienst vorgenommen wurden. Abonnieren Sie den RSS Feed auf der Seite AWS PCS Dokumentenverlauf, um automatische Benachrichtigungen über Änderungen an dieser Seite zu erhalten.

Änderung	Beschreibung	Datum
AWS PCS hat die Änderungs verfolgung gestartet	AWS PCShat begonnen, Änderungen für die AWS verwalteten Richtlinien zu verfolgen.	28. August 2024

Serviceverknüpfte Rollen für AWS PCS

AWS Parallel Computing Service verwendet AWS Identity and Access Management (IAM) dienstverknüpfte Rollen. Eine dienstbezogene Rolle ist ein einzigartiger Rollentyp, mit dem direkt verknüpft ist. IAM AWS PCS Mit Diensten verknüpfte Rollen sind vordefiniert AWS PCS und enthalten alle Berechtigungen, die der Dienst benötigt, um andere AWS Dienste in Ihrem Namen aufzurufen.

Eine dienstbezogene Rolle AWS PCS erleichtert die Einrichtung, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. AWS PCSdefiniert die Berechtigungen ihrer dienstbezogenen Rollen und AWS PCS kann, sofern nicht anders definiert, nur ihre Rollen übernehmen. Zu den definierten Berechtigungen gehören die Vertrauensrichtlinie und die Berechtigungsrichtlinie, und diese Berechtigungsrichtlinie kann keiner anderen IAM Entität zugeordnet werden.

Sie können eine serviceverknüpfte Rolle erst löschen, nachdem die zugehörigen Ressourcen gelöscht wurden. Dadurch werden Ihre AWS PCS Ressourcen geschützt, da Sie die Zugriffsberechtigung für die Ressourcen nicht versehentlich entfernen können.

Informationen zu anderen Diensten, die dienstbezogene Rollen unterstützen, finden Sie unter AWS Dienste, die mit Diensten arbeiten, IAM und suchen Sie in der Spalte Dienstbezogene Rollen nach

den Diensten, für die Ja angegeben ist. Wählen Sie über einen Link Ja aus, um die Dokumentation zu einer serviceverknüpften Rolle für diesen Service anzuzeigen.

Berechtigungen für dienstverknüpfte Rollen für AWS PCS

AWS PCSverwendet die serviceverknüpfte Rolle mit dem Namen AWSServiceRoleForPCS—Erlaube AWS PCS die Verwaltung von EC2 Amazon-Ressourcen.

Die AWSServiceRoleForPCS servicebezogene Rolle vertraut darauf, dass die folgenden Dienste die Rolle übernehmen:

- `pcs.amazonaws.com`

Die genannte Richtlinie für Rollenberechtigungen AWSPCSServiceRolePolicyermöglicht AWS PCS das Ausführen von Aktionen für bestimmte Ressourcen.

Sie müssen Berechtigungen konfigurieren, damit eine Benutzer, Gruppen oder Rollen eine serviceverknüpfte Rolle erstellen, bearbeiten oder löschen können. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Dienstbezogene Rollenberechtigungen.

Erstellen einer dienstbezogenen Rolle für AWS PCS

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. AWS PCSerstellt für Sie eine dienstverknüpfte Rolle, wenn Sie einen Cluster erstellen.

Bearbeiten einer serviceverknüpften Rolle für AWS PCS

AWS PCSerlaubt es Ihnen nicht, die AWSServiceRoleForPCS dienstbezogene 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 die Beschreibung der Rolle jedoch mithilfe IAM von bearbeiten. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Bearbeiten einer dienstbezogenen Rolle.

Löschen einer serviceverknüpften Rolle für AWS PCS

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 verwendet werden von AWSServiceRoleForPCS

Sie müssen alle Ihre Cluster löschen, um die AWSServiceRoleForPCS dienstverknüpfte Rolle zu löschen. Weitere Informationen finden Sie unter Löschen eines Clusters.

Um die mit dem Service verknüpfte Rolle manuell zu löschen, verwenden Sie IAM

Verwenden Sie die IAM Konsole, den oder AWS CLI, AWS API um die AWSServiceRoleForPCS dienstverknüpfte Rolle zu löschen. Weitere Informationen finden Sie im IAMBenutzerhandbuch unter Löschen einer dienstbezogenen Rolle.

Unterstützte Regionen für serviceverknüpfte AWS PCS-Rollen

AWS PCSunterstützt die Verwendung von dienstbezogenen Rollen in allen Regionen, in denen der Dienst verfügbar ist. Weitere Informationen finden Sie unter AWS Regionen und Endpunkte.

Amazon EC2 Spot-Rolle für AWS PCS

Wenn Sie eine AWS PCS Compute-Knotengruppe erstellen möchten, die Spot als Kaufoption verwendet, müssen Sie auch die AWSServiceRoleForEC2Spotserviceverknüpfte Rolle in Ihrer AWS-Konto haben. 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 PCS

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 API Aktionen

APIAktion	Mindestberechtigungen	Zusätzliche Berechtigungen für die Konsole
CreateCluster	`ec2:CreateNetworkInterface, ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:GetSecurityGroupsForVpc, iam:CreateServiceLinkedRole, secretsmanager:CreateSecret, secretsmanager:TagResource, pcs:CreateCluster`	

APIAktion	Mindestberechtigungen	Zusätzliche Berechtigungen für die Konsole
ListClusters	`pcs:ListClusters`	
GetCluster	`pcs:GetCluster`	`ec2:DescribeSubnets`
DeleteCluster	`pcs:DeleteCluster`	
CreateComputeNodeGroup	`ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:DescribeLaunchTemplates, ec2:DescribeLaunchTemplateVersions, ec2:DescribeInstanceTypes, ec2:RunInstances, ec2:CreateFleet, ec2:CreateTags, iam:PassRole, iam:GetInstanceProfile, pcs:CreateComputeNodeGroup`	`iam:ListInstanceProfiles, ec2:DescribeImages, pcs:GetCluster`
ListComputerNodeGroups	`pcs:ListComputeNodeGroups`	`pcs:GetCluster`
GetComputeNodeGroup	`pcs:GetComputeNodeGroup`	`ec2:DescribeSubnets`

APIAktion	Mindestberechtigungen	Zusätzliche Berechtigungen für die Konsole
UpdateComputeNodeGroup	`ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:DescribeLaunchTemplates, ec2:DescribeLaunchTemplateVersions, ec2:DescribeInstanceTypes, 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`	
UpdateQueue	`pcs:UpdateQueue`	`pcs:ListComputeNodeGroups, pcs:GetQueue`

APIAktion	Mindestberechtigungen	Zusätzliche Berechtigungen für die Konsole
DeleteQueue	`pcs:DeleteQueue`	

Für die Verwendung von Tags sind Mindestberechtigungen erforderlich

Die folgenden Berechtigungen sind erforderlich, um Tags mit Ihren Ressourcen verwenden zu können AWS PCS.

```
pcs:ListTagsForResource
pcs:TagResource
pcs:UntagResource
```

Für die Unterstützung von Protokollen sind Mindestberechtigungen erforderlich

AWS PCSsendet 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 für einen Service-Administrator

Die folgende IAM Richtlinie legt die Mindestberechtigungen fest, die für eine IAM Identität (Benutzer, Gruppe oder Rolle) erforderlich sind, um den AWS PCS Dienst 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 fürSSH.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface",
                "ec2:DescribeImages",
                "ec2:DescribeInstanceTypes",
                "ec2:DescribeLaunchTemplates",
                "ec2:DescribeLaunchTemplateVersions",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:GetSecurityGroupsForVpc",
                "firehose:*",
                "iam:GetInstanceProfile",
                "iam:ListInstanceProfiles",
                "iam:PassRole",
                "kms:*",
                "logs:*",
                "pcs:*",
                "s3:*"
            ],
            "Resource": "*"
        }
    ]
}
```

Sie können die folgenden Berechtigungen aus der Richtlinie ausschließen und stattdessen die entsprechende verwaltete Richtlinie verwenden inIAM:

- `"firehose:*"`

  ```
  AmazonKinesisFirehoseFullAccess
  ```

- `"kms:*"`

  ```
  AWSKeyManagementServicePowerUser
  ```

- `"logs:*"`

```
CloudWatchLogsFullAccess
```

- `"s3:*"`

```
AmazonS3FullAccess
```

IAMInstanzprofile für AWS Parallel Computing Service

Anwendungen, die auf einer EC2 Instanz ausgeführt werden, müssen in allen AWS API Anfragen, die sie stellen, AWS Anmeldeinformationen enthalten. Wir empfehlen, eine IAM Rolle zu verwenden, um temporäre Anmeldeinformationen auf der EC2 Instanz zu verwalten. Sie können zu diesem Zweck ein Instanzprofil definieren und es an Ihre Instances anhängen. Weitere Informationen finden Sie unter IAMRollen für Amazon EC2 im Amazon Elastic Compute Cloud-Benutzerhandbuch.

> **Note**
>
> Wenn Sie die verwenden, AWS Management Console um eine IAM Rolle für Amazon zu erstellenEC2, erstellt die Konsole automatisch ein Instance-Profil und weist diesem den gleichen Namen wie die IAM Rolle zu. Wenn Sie zum Erstellen der IAM Rolle die AWS API Aktionen AWS CLI, oder AWS SDK an verwenden, erstellen Sie das Instance-Profil als separate Aktion. Weitere Informationen finden Sie unter Instanzprofile im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Sie müssen das ARN eines Instance-Profils angeben, wenn Sie Compute-Knotengruppen erstellen. Sie können verschiedene Instanzprofile für einige oder alle Compute-Knotengruppen auswählen.

Anforderungen an das Instanzprofil

Name des Instanzprofils

Das IAM Instanzprofil ARN muss entweder mit dem Pfad beginnen `AWSPCS` oder diesen enthalten`/aws-pcs/`.

Example

- `arn:aws:iam::*:instance-profile/AWSPCS-example-role-1` und

- `arn:aws:iam::*:instance-profile/aws-pcs/example-role-2`.

Berechtigungen

Das Instanzprofil für AWS PCS muss mindestens die folgende Richtlinie enthalten. Es ermöglicht Rechenknoten, den AWS PCS Dienst zu benachrichtigen, wenn sie betriebsbereit sind.

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

Zusätzliche Richtlinien

Sie könnten erwägen, verwaltete Richtlinien zum Instanzprofil hinzuzufügen. Beispielsweise:

- AmazonS3 ReadOnlyAccess bietet schreibgeschützten Zugriff auf alle S3-Buckets.
- A mazonSSMManaged InstanceCore aktiviert 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 eines Instance-Profils

Sie können ein Instance-Profil direkt von der EC2 Amazon-Konsole aus erstellen. Weitere Informationen finden Sie unter Verwenden von Instance-Profilen im AWS Identity and Access Management Benutzerhandbuch.

Problembehandlung bei Identität und Zugriff auf den AWS 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 auftreten könnenIAM.

Themen

Ich bin nicht berechtigt, eine Aktion durchzuführen in AWS PCS

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 `mateojackson` IAM Benutzer versucht, die Konsole zu verwenden, um Details zu einer fiktiven *`my-example-widget`* Ressource anzuzeigen, aber nicht über die fiktiven `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 eine Fehlermeldung erhalten, dass Sie nicht berechtigt sind, die `iam:PassRole` Aktion auszuführen, müssen Ihre Richtlinien aktualisiert werden, damit Sie eine Rolle an AWS PCS diese Person 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 von mir den Zugriff AWS-Konto auf meine AWS PCS Ressourcen ermöglichen

Sie können eine Rolle erstellen, die Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation für den Zugriff auf Ihre Ressourcen verwenden können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Für Dienste, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (ACLs) unterstützen, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob diese Funktionen AWS PCS unterstützt werden, finden Sie unter. So funktioniert AWS Parallel Computing Service 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 auf einen anderen IAMBenutzer AWS-Konto , der Ihnen gehört.
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie AWS-Konten im IAMBenutzerhandbuch unter Gewähren des Zugriffs für Dritte.
- Informationen dazu, wie Sie Zugriff über einen Identitätsverbund gewähren, finden Sie im Benutzerhandbuch unter Zugriff für extern authentifizierte Benutzer (Identitätsverbund). IAM
- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontenübergreifenden Zugriff finden Sie IAMim Benutzerhandbuch unter Kontoübergreifender Ressourcenzugriff. IAM

Konformitätsprüfung für Parallel Computing Service AWS

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. AWS stellt die folgenden Ressourcen zur Verfügung, die Sie bei der Einhaltung der Vorschriften unterstützen:

- Schnellstartanleitungen zu Sicherheit und Compliance — In diesen Bereitstellungsleitfäden werden architektonische Überlegungen erörtert und Schritte für die Implementierung von Basisumgebungen beschrieben AWS , bei denen Sicherheit und Compliance im Mittelpunkt stehen.
- Architecting for HIPAA Security and Compliance on Amazon Web Services — In diesem Whitepaper wird beschrieben, wie Unternehmen Anwendungen erstellen HIPAA können, die AWS für sie in Frage kommen.

  > **Note**
  >
  > Nicht alle sind berechtigt AWS-Services . HIPAA Weitere Informationen finden Sie in der Referenz für HIPAA qualifizierte Dienste.

- AWS Ressourcen zur AWS von Vorschriften — Diese Sammlung von Arbeitsmappen und Leitfäden kann auf Ihre Branche und Ihren Standort zutreffen.
- AWS Leitfäden zur Einhaltung von Vorschriften für Kunden — Verstehen Sie das Modell der gemeinsamen Verantwortung aus dem Blickwinkel der Einhaltung von Vorschriften. In den Leitfäden werden die bewährten Verfahren zur Sicherung zusammengefasst AWS-Services und die Leitlinien für Sicherheitskontrollen in verschiedenen Frameworks (einschließlich des National Institute of Standards and Technology (NIST), des Payment Card Industry Security Standards Council (PCI) und der International Organization for Standardization (ISO)) zusammengefasst.
- Evaluierung von Ressourcen anhand von Regeln im AWS Config Entwicklerhandbuch — Der AWS Config Service bewertet, wie gut Ihre Ressourcenkonfigurationen den internen Praktiken, Branchenrichtlinien und Vorschriften entsprechen.

- AWS Security Hub— Auf diese AWS-Service Weise erhalten Sie einen umfassenden Überblick über Ihren internen Sicherheitsstatus. AWS Security Hub verwendet Sicherheitskontrollen, um Ihre AWS -Ressourcen zu bewerten und Ihre Einhaltung von Sicherheitsstandards und bewährten Methoden zu überprüfen. Eine Liste der unterstützten Services und Kontrollen finden Sie in der Security-Hub-Steuerungsreferenz.
- Amazon GuardDuty — Dies AWS-Service erkennt potenzielle Bedrohungen für Ihre Workloads AWS-Konten, Container und Daten, indem es Ihre Umgebung auf verdächtige und böswillige Aktivitäten überwacht. GuardDuty kann Ihnen helfen, verschiedene Compliance-Anforderungen zu erfüllen PCIDSS, z. B. durch die Erfüllung der Anforderungen zur Erkennung von Eindringlingen, die in bestimmten Compliance-Frameworks vorgeschrieben sind.
- AWS Audit Manager— Auf diese AWS-Service Weise können Sie Ihre AWS Nutzung kontinuierlich überprüfen, um das Risikomanagement und die Einhaltung von Vorschriften und Industriestandards zu vereinfachen.

Ausfallsicherheit im AWS Parallel-Computing-Service

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.

Infrastruktursicherheit im AWS Parallel Computing Service

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 für den Zugriff AWS PCS über das Netzwerk. Kunden müssen Folgendes unterstützen:

- Sicherheit auf Transportschicht (TLS). Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Cipher-Suites mit Perfect Forward Secrecy (PFS) wie (Ephemeral Diffie-Hellman) oder DHE (Elliptic Curve Ephemeral Diffie-Hellman). ECDHE Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi.

Darüber hinaus müssen Anfragen mithilfe einer Zugriffsschlüssel-ID und eines geheimen Zugriffsschlüssels, der einem Prinzipal zugeordnet ist, signiert werden. IAM Alternativ können Sie mit AWS Security Token Service (AWS STS) temporäre Sicherheitsanmeldeinformationen erstellen, um die Anforderungen zu signieren.

Wenn ein Cluster AWS PCS erstellt wird, 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, AWS PCS erstellt er ein kontoübergreifendes Elastic Network Interface (ENI) in Ihrem. VPC Der Slurm-Controller verwendet denENI, 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 KI/ML-Operationen HPC ermöglicht werden.

Analyse und Verwaltung von Sicherheitslücken im Parallel Computing Service AWS

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.

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 auf Knotengruppen-Instances.
- Aktualisieren Sie den Scheduler, damit er immer innerhalb der unterstützten Versionen bleibt.
- Authentifizieren und verschlüsseln Sie die Kommunikation zwischen Benutzerclients und den Knoten, mit denen sie sich verbinden.

Serviceübergreifende Confused-Deputy-Prävention

Das Confused-Deputy-Problem ist ein Sicherheitsproblem, bei dem eine juristische Stelle, die nicht über die Berechtigung zum Ausführen einer Aktion verfügt, eine privilegiertere juristische Stelle zwingen kann, die Aktion auszuführen. Im Fall AWS eines dienstübergreifenden Identitätswechsels kann das Problem des verwirrten Stellvertreters auftreten. Ein dienstübergreifender Identitätswechsel kann auftreten, wenn ein Dienst (der Anruf-Dienst) einen anderen Dienst anruft (den aufgerufenen Dienst). 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, sich vor dem Problem mit dem verwirrten Deputy zu schützen, besteht darin, den `aws:SourceArn` globalen Bedingungskontextschlüssel mit ARN der gesamten Ressource zu verwenden. Wenn Sie die gesamte ARN Ressource nicht kennen oder wenn Sie mehrere Ressourcen angeben, verwenden Sie den `aws:SourceArn` globalen Kontextbedingungsschlüssel mit Platzhalterzeichen (*) für die unbekannten Teile von. ARN Beispiel, `arn:aws:`*`servicename`*`:*:`*`123456789012`*`:*`.

Wenn der `aws:SourceArn` Wert die Konto-ID nicht enthält, z. B. ein Amazon S3 S3-BucketARN, müssen Sie beide globalen Bedingungskontextschlüssel verwenden, um die Berechtigungen einzuschränken.

Der Wert von `aws:SourceArn` muss ein Cluster seinARN.

Das folgende Beispiel zeigt, wie Sie die Kontextschlüssel `aws:SourceArn` und die `aws:SourceAccount` globale Bedingung verwenden können, AWS PCS 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"
      }
    }
  }
}
```

IAMRolle für EC2 Amazon-Instances, die als Teil einer Compute-Knotengruppe bereitgestellt werden

AWS PCSorchestriert automatisch die EC2 Amazon-Kapazität für jede der konfigurierten Rechenknotengruppen in einem Cluster. Bei der Erstellung einer Rechenknotengruppe müssen Benutzer über das `iamInstanceProfileArn` Feld ein IAM Instanzprofil angeben. Das Instanzprofil gibt die Berechtigungen an, die den bereitgestellten EC2 Instanzen zugeordnet sind. AWS PCSakzeptiert jede Rolle, die ein Rollennamenpräfix hat `AWSPCS` oder `/aws-pcs/` die Teil des Rollenpfads ist. Die `iam:PassRole` Berechtigung ist für die IAM Identität (Benutzer oder Rolle) erforderlich, die eine Rechenknotengruppe erstellt oder aktualisiert. Wenn ein Benutzer die `CreateComputeNodeGroup` `UpdateComputeNodeGroup` API Oder-Aktionen aufruft, wird AWS PCS geprüft, ob der Benutzer die `iam:PassRole` Aktion ausführen darf.

Die folgende Beispielrichtlinie gewährt nur Berechtigungen zur Weitergabe von IAM Rollen, deren Name mit beginnt`AWSPCS`.

```
{
    "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 Parallel Computing Service

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

AMIverwandte Sicherheit

- Verwenden Sie AWS PCS Sample nicht AMIs für Produktionsworkloads. Die Beispiele AMIs werden nicht unterstützt und sind nur zum Testen bestimmt.
- Aktualisieren Sie regelmäßig das Betriebssystem und die Software der AWS PCS Instances, um Sicherheitslücken zu beheben.
- Verwenden Sie AWS Systems Manager es, um das Patchen zu automatisieren und die Einhaltung Ihrer Sicherheitsrichtlinien zu gewährleisten.
- Verwenden Sie nur authentifizierte offizielle AWS PCS Pakete, die von offiziellen AWS Quellen heruntergeladen wurden.
- Aktualisieren Sie regelmäßig AWS PCS Pakete auf Rechenknoten, um Sicherheitspatches und Verbesserungen zu erhalten. Erwägen Sie, diesen Prozess zu automatisieren, um Sicherheitslücken zu minimieren.

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 AWS PCS RSS Dokumentations-Feed zu abonnieren und eine Benachrichtigung zu erhalten, wenn sich eine Slurm-Version nähert. EOSL

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 einem separaten Bereich bereitVPC, um Ihre HPC Umgebung von anderem Netzwerkverkehr zu isolieren.
- Verwenden Sie Sicherheitsgruppen und Netzwerkzugriffskontrolllisten (ACLs), um den ein- und ausgehenden Datenverkehr zu AWS PCS Instances und Subnetzen zu kontrollieren.
- Verwenden Sie AWS PrivateLink unsere VPC Endpunkte, um den Netzwerkverkehr zwischen Ihren Clustern und anderen AWS Diensten innerhalb des Netzwerks aufrechtzuerhalten. AWS

Protokollierung und Überwachung für AWS PCS

Die Überwachung ist ein wichtiger Bestandteil der Aufrechterhaltung der Zuverlässigkeit, Verfügbarkeit und Leistung Ihrer AWS PCS anderen AWS Ressourcen. AWSbietet die folgenden Überwachungstools, mit denen Sie beobachten AWS PCS, melden können, wenn etwas nicht stimmt, und gegebenenfalls automatische Maßnahmen ergreifen können:

- 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 Nutzung oder andere Kennzahlen Ihrer EC2 Amazon-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 EC2 Amazon-Instances und anderen Quellen überwachen CloudTrail, speichern und darauf zugreifen. CloudWatch Logs kann 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 Anrufe 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 Anrufe erfolgten. Weitere Informationen finden Sie im AWS CloudTrail -Benutzerhandbuch.

AWS PCSScheduler-Protokolle

Sie können so konfigurieren AWS PCS, dass detaillierte Protokollierungsdaten 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. Sie können AWS PCS Scheduler-Protokolle sowohl mit der AWS PCS Konsole als auch programmgesteuert mit dem oder einrichten. AWS CLI SDK

Inhalt

Voraussetzungen

Der zur Verwaltung des AWS PCS Clusters verwendete IAM Prinzipal muss dies zulassen. `pcs:AllowVendedLogDeliveryForResource` Im Folgenden finden Sie ein Beispiel AWS IAM für eine Richtlinie, mit der dies aktiviert wird.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PcsAllowVendedLogsDelivery",
            "Effect": "Allow",
            "Action": ["pcs:AllowVendedLogDeliveryForResource"],
            "Resource": [
                "arn:aws:pcs:::cluster/*"
            ]
        }
    ]
}
```

Scheduler-Logs mithilfe der AWS PCS Konsole einrichten

Gehen Sie wie folgt vor, um AWS PCS Scheduler-Logs in der Konsole einzurichten:

1. Öffnen Sie die AWS PCSKonsole.
2. Wählen Sie Cluster und navigieren Sie zur Detailseite für den AWS PCS Cluster, auf der Sie die Protokollierung aktivieren möchten.
3. Wählen Sie Logs.
4. Unter Protokolllieferungen — Scheduler Logs — optional

a. Fügen Sie bis zu drei Ziele für die Protokollzustellung hinzu. Zur Auswahl stehen CloudWatch Logs, Amazon S3 oder Firehose.

b. Wählen Sie Protokolllieferungen aktualisieren aus.

Sie können Protokollzustellungen neu konfigurieren, hinzufügen oder entfernen, indem Sie diese Seite erneut aufrufen.

Einrichten von Scheduler-Protokollen mit dem AWS CLI

Um dies zu erreichen, benötigen Sie mindestens ein Zustellungsziel, eine Zustellungsquelle (den PCS Cluster) und eine Zustellung, bei der es sich um eine Beziehung handelt, die eine Quelle mit einem Ziel verbindet.

Erstellen Sie ein Lieferziel

Sie benötigen mindestens ein Lieferziel, um Scheduler-Protokolle von einem AWS PCS Cluster zu empfangen. Weitere Informationen zu diesem Thema finden Sie im PutDeliveryDestination Abschnitt des CloudWatch API Benutzerhandbuchs.

Um ein Lieferziel zu erstellen, verwenden Sie AWS CLI

- Erstellen Sie ein Ziel mit dem folgenden Befehl. Nehmen Sie vor der Ausführung des Befehls die folgenden Ersetzungen vor:
  - Ersetzen *region-code* mit dem AWS-Region Ort, an dem Sie Ihr Ziel erstellen werden. In der Regel handelt es sich dabei um dieselbe Region, in der der AWS PCS Cluster bereitgestellt wird.
  - Ersetzen *pcs-logs-destination* mit Ihrem bevorzugten Namen. Es muss für alle Lieferziele in Ihrem Konto eindeutig sein.
  - Ersetzen *resource-arn* mit dem ARN für eine bestehende Protokollgruppe in CloudWatch Logs, einem S3-Bucket oder einem Lieferstream in Firehose. Beispiele sind unter anderem:
    - CloudWatch Gruppe „Protokolle“

      ```
      arn:aws:logs:region-code:account-id:log-group:/log-group-name:*
      ```

    - S3 bucket

      ```
      arn:aws:s3:::bucket-name
      ```

    - Firehose-Lieferstrom

```
arn:aws:firehose:region-code:account-id:deliverystream/stream-name
```

```
aws logs put-delivery-destination --region region-code \
    --name pcs-logs-destination \
    --delivery-destination-configuration destinationResourceArn=resource-arn
```

Notieren Sie sich den ARN für das neue Lieferziel, da Sie ihn zur Konfiguration von Lieferungen benötigen.

Aktivieren Sie den AWS PCS Cluster als Zustellungsquelle

Um Scheduler-Protokolle zu sammeln AWSPCS, konfigurieren Sie den Cluster als Bereitstellungsquelle. Weitere Informationen finden Sie PutDeliverySourcein der Amazon CloudWatch API Logs-Referenz.

Um einen Cluster als Bereitstellungsquelle zu konfigurieren, verwenden Sie AWS CLI

- Aktivieren Sie die Protokollzustellung von Ihrem Cluster aus mit dem folgenden Befehl. Nehmen Sie vor der Ausführung des Befehls die folgenden Ersetzungen vor:
  - Ersetzen *region-code* mit dem AWS-Region Ort, an dem Ihr Cluster bereitgestellt wird.
  - Ersetzen *cluster-logs-source-name* mit einem Namen für diese Quelle. Es muss für alle Lieferquellen in Ihrer eindeutig sein AWS-Konto. Erwägen Sie, den Namen oder die ID des AWS PCS Clusters einzubeziehen.
  - Ersetzen *cluster-arn* mit dem ARN für Ihren Cluster AWS PCS

```
aws logs put-delivery-source \
    --region region-code \
    --name cluster-logs-source-name \
    --resource-arn cluster-arn \
    --log-type PCS_SCHEDULER_LOGS
```

Connect die Cluster-Bereitstellungsquelle mit dem Übermittlungsziel

Damit Scheduler-Protokolldaten vom Cluster zum Ziel fließen können, müssen Sie eine Bereitstellung konfigurieren, die sie verbindet. Weitere Informationen finden Sie CreateDeliveryin der Amazon CloudWatch API Logs-Referenz.

Um eine Lieferung mit dem zu erstellen AWS CLI

- Erstellen Sie eine Lieferung mit dem folgenden Befehl. Nehmen Sie vor der Ausführung des Befehls die folgenden Ersetzungen vor:
  - Ersetzen *region-code* mit dem AWS-Region Ort, an dem sich Ihre Quelle und Ihr Ziel befinden.
  - Ersetzen *cluster-logs-source-name* mit dem Namen Ihrer Lieferquelle von oben.
  - Ersetzen *destination-arn* mit dem ARN von einem Lieferziel, an das die Logs geliefert werden sollen.

```
aws logs create-delivery \
    --region region-code \
    --delivery-source-name cluster-logs-source \
    --delivery-destination-arn destination-arn
```

Pfade und Namen der Protokolldatenströme im Scheduler

Der Pfad und der Name für AWS PCS Scheduler-Protokolle hängen vom Zieltyp ab.

- CloudWatch Protokolle
  - Ein CloudWatch Logs-Stream folgt dieser Namenskonvention.

    ```
    AWSLogs/PCS/${cluster_id}/${log_name}_${scheduler_major_version}.log
    ```

    Example

    ```
    AWSLogs/PCS/abcdef0123/slurmctld_24.05.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/24.05/2024/09/01/00.
    ```

- 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_24.05_2024-09-01-00_abcdef0123_0123abcdef.log
```

Beispiel für einen AWS PCS Scheduler-Protokolleintrag

AWSPCSScheduler-Protokolle sind strukturiert. Sie enthalten Felder wie die Cluster-ID, den Scheduler-Typ, Haupt- und Patch-Versionen sowie die Protokollnachricht, die vom Slurm-Controller-Prozess ausgegeben wird. Ein Beispiel.

```
{
    "resource_id": "s3431v9rx2",
    "resource_type": "PCS_CLUSTER",
    "event_timestamp": 1721230979,
    "log_level": "info",
    "log_name": "slurmctld",
    "scheduler_type": "slurm",
    "scheduler_major_version": "23.11",
    "scheduler_patch_version": "8",
    "node_type": "controller_primary",
    "message": "[2024-07-17T15:42:58.614+00:00] Running as primary controller\n"
}
```

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 Intervallen Metriken aus dem Cluster sammelt. Diese Metriken werden beibehalten, 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 EC2 Instances zu überwachen, die von gestartet wurden AWS PCS, um Ihre Skalierungsanforderungen zu erfüllen. Sie können zwar die Protokolle

laufender Instances überprüfen, aber CloudWatch Metriken und Protokolldaten werden 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 AWS PCS Verwendung CloudWatch von Monitoring zu erfahren.

Themen

AWS PCSMetriken überwachen mit CloudWatch

Sie können den Zustand Ihres 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 PCSveröffentlicht die folgenden Metriken im PCS Namespace AWS/in CloudWatch. Sie haben eine einzige Dimension,`ClusterId`.

Name	Beschreibung	Einheiten
ActualCapacity	IdleCapacity + UtilizedCapacity	Anzahl
CapacityUtilization	UtilizedCapacity / ActualCapacity	Anzahl
DesiredCapacity	ActualCapacity + PendingCapacity	Anzahl

Name	Beschreibung	Einheiten
IdleCapacity	Anzahl der Instanzen, die ausgeführt werden, aber keinen Jobs zugewiesen sind	Anzahl
UtilizedCapacity	Anzahl der Instanzen, die ausgeführt werden und Jobs zugewiesen sind	Anzahl

AWS PCSInstances mithilfe von Amazon überwachen CloudWatch

AWSPCSstartet EC2 Amazon-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-Logs werden in der Regel zusammen mit den EBS Volumes gelöscht, die die Instance unterstützen. Um Metriken oder Protokolldaten der Instances beizubehalten, die PCS nach deren Beendigung gestartet wurden, 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

AWSPCSInstanzen finden

Um Instances zu überwachen, die von gestartet wurdenPCS, suchen Sie nach den laufenden Instances, die einem Cluster oder einer Compute-Knotengruppe zugeordnet sind. Überprüfen Sie dann in der EC2 Konsole für eine bestimmte Instanz 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 einsehen. Weitere Informationen zur Identifizierung der Instanzen, von denen verwaltet wirdPCS, finden Sie unterSuchen nach Instanzen der Compute-Knotengruppe 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 undVerwenden von EC2 Amazon-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 mit einem Texteditor erstellt werden. Die Konfigurationsdatei wird für diese Demonstration manuell erstellt.

Erstellen Sie auf einem Computer, auf dem Sie die AWS CLI installiert haben, eine CloudWatch Konfigurationsdatei namens config.json mit dem folgenden Inhalt. Sie können auch Folgendes verwendenURL, 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 2. Wenn Ihre Instances ein anderes Basisbetriebssystem verwenden, ändern Sie die Pfade entsprechend.

- Um andere Logs 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 -spezifischen 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 `/PCSLogs/instances` gesendet. 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 Rechenknoteninstanzen zugegriffen werden kann. Es gibt zwei gängige Methoden, dies zu tun. 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. Bevor Sie den Befehl ausführen, nehmen Sie folgende Ersetzungen vor:

- Ersetzen *DOC-EXAMPLE-BUCKET* mit Ihrem 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://DOC-EXAMPLE-BUCKET
```

Laden Sie als Nächstes die Datei in den Bucket hoch.

```
aws s3 cp ./config.json s3://DOC-EXAMPLE-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:

- Ersetzen *region-code* mit der AWS Region, in der Sie arbeiten AWSPCS.
- (Optional) Ersetzen *AmazonCloudWatch-PCS* durch Ihren eigenen Namen für den Parameter. 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 gespeichert ist oderSSM.

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:

- *DOC-EXAMPLE-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://DOC-EXAMPLE-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 eine IAM Beispielrichtlinie für den Bucket im obigen Benutzerdatenskript.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET",
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ]
        }
    ]
}
```

Beachten Sie außerdem, dass die Instances ausgehenden Datenverkehr zum S3 und zu den CloudWatch Endpunkten zulassen müssen. Dies kann je nach Ihrer Cluster-Architektur mithilfe von Sicherheitsgruppen oder VPC Endpunkten erreicht werden.

Verwenden Sie eine Konfiguration, die in gespeichert ist SSM

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 Daten:

- (Optional) Ersetzen *AmazonCloudWatch-PCS* durch Ihren eigenen Namen für den Parameter.

```
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*.

```
{
  "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 zu den CloudWatch Endpunkten SSM und zulassen müssen. Dies kann je nach Clusterarchitektur mithilfe von Sicherheitsgruppen oder VPC Endpunkten erreicht werden.

Protokollieren von AWS Parallel Computing API Service-Aufrufen mit AWS CloudTrail

AWS PCSist in einen Dienst integriert AWS CloudTrail, der eine Aufzeichnung der Aktionen bereitstellt, die von einem Benutzer, einer Rolle oder einem AWS Dienst in ausgeführt wurden AWS PCS. CloudTrail erfasst alle API Aufrufe AWS PCS als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der AWS PCS Konsole und Code-Aufrufe der AWS PCS API Operationen. Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Bereitstellung 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 ermitteln CloudTrail, an die die Anfrage gestellt wurde AWS PCS, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere Details.

Weitere Informationen CloudTrail dazu finden Sie im AWS CloudTrail Benutzerhandbuch.

AWS PCSInformationen in CloudTrail

CloudTrail ist auf Ihrem aktiviert AWS-Konto , wenn Sie das Konto erstellen. Wenn eine Aktivität in stattfindet AWS PCS, wird diese Aktivität zusammen mit anderen AWS Serviceereignissen in der CloudTrail Ereignishistorie 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 AWS-Konto, einschließlich der 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 SNS Amazon-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 Reference 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 im CloudTrail userIdentityElement.

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 PCS

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 Dienstressourcen oder Vorgängen für Ihren AWS-Konto.

Ihr AWS-Konto hat Standardkontingente für jeden AWS Dienst. Wenn nicht anders angegeben, gilt jedes Kontingent spezifisch für eine Region. 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	Protokoll
USA Ost (Nord-Virginia)	us-east-1	pcs.us-east-1.amazonaws.com	HTTPS
USA Ost (Ohio)	us-east-2	pcs.us-east-2.amazonaws.com	HTTPS
USA West (Oregon)	us-west-2	pcs.us-west-2.amazonaws.com	HTTPS
Asien-Pazifik (Singapur)	ap-southeast-1	pcs.ap-southeast-1.amazonaws.com	HTTPS
Asien-Pazifik (Sydney)	ap-southeast-2	pcs.ap-southeast-2.amazonaws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
Asien-Pazifik (Tokio)	ap-northeast-1	pcs.ap-northeast-1.amazonaws.com	HTTPS
Europa (Frankfurt)	eu-central-1	pcs.eu-central-1.amazonaws.com	HTTPS
Europa (Irland)	eu-west-1	pcs.eu-west-1.amazonaws.com	HTTPS
Europa (Stockholm)	eu-north-1	pcs.eu-north-1.amazonaws.com	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 Management Console. 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 Management Console.

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

Relevante Kontingente für andere AWS Dienste

AWS PCSnutzt andere AWS Dienste. Ihre Dienstkontingente für diese Dienste wirken sich auf Ihre Nutzung von aus AWS PCS.

EC2Amazon-Servicekontingente, die sich auswirken AWS PCS

- Spot-Instance-Anfragen
- On-Demand-Instances ausführen
- Startvorlagen
- Startvorlagenversionen
- EC2APIAmazon-Anfragen

Weitere Informationen finden Sie unter Amazon EC2 Service Quotas im Amazon Elastic Compute Cloud-Benutzerhandbuch.

Versionshinweise für ein AWS PCS Beispiel AMIs

AWS PCSBeispiele AMIs haben einen nächtlichen Veröffentlichungsrhythmus für Sicherheitspatches. Diese inkrementellen Sicherheitspatches sind nicht in den offiziellen Versionshinweisen enthalten.

> **Important**
>
> AMIsDie Beispiele dienen zu Demonstrationszwecken und werden nicht für Produktionsworkloads empfohlen.

Inhalt

AWS PCSBeispiel x86_64 AMI für Slurm 23.11 (Amazon Linux 2)

Dieses Dokument beschreibt die neuesten Änderungen, Ergänzungen, bekannten Probleme und Korrekturen für AWS PCS Sample x86_64 AMI (Amazon Linux 2).

- Erstellungsdatum: 15. Juli 2024
- Datum der Veröffentlichung: 22. August 2024
- Letzte Aktualisierung: 22. August 2024

AMIName

- `aws-pcs-sample_ami-amzn2-x86_64-slurm-23.11`

Unterstützte EC2 Instanzen

- Alle Instanzen mit einem 64-Bit-x86-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur EC2Amazon-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach`Architectures=x86_64`.

AMIInhalt

- Unterstützter AWS Dienst: AWS PCS
- Betriebssystem: Amazon Linux 2
- Rechenarchitektur: x86_64
- Linux-Kernel: 5.10.220-209.867.amzn2.x86_64
- EBSDatenträgertyp: gp2
- AWS PCSSlurm 23.11-Installationsprogramm: 23.11.9-1
- AWS PCSSoftwareinstallationsprogramm: 1.0.0-1
- EFAInstallationsprogramm: 1.33.0
- GDRCopy: 2.4
- NVIDIAFahrer: 535.154.05
- NVIDIACUDA: 12.2.2_535.104.05

Hinweise

- None

Veröffentlichungsdatum: 2024-08-22

Aktualisiert

- Keine. Erste Veröffentlichung.

Hinzugefügt

- Keine. Erste Veröffentlichung.

Entfernt

- Keine. Erste Veröffentlichung.

AWS PCSBeispiel Arm64 AMI für Slurm 23.11 (Amazon Linux 2)

Dieses Dokument beschreibt die neuesten Änderungen, Ergänzungen, bekannten Probleme und Korrekturen für AWS PCS Sample Arm64 AMI (Amazon Linux 2).

- Erstellungsdatum: 15. Juli 2024
- Datum der Veröffentlichung: 22. August 2024
- Letzte Aktualisierung: 22. August 2024

AMIName

- `aws-pcs-sample_ami-amzn2-arm64-slurm-23.11`

Unterstützte EC2 Instanzen

- Alle Instanzen mit einem 64-Bit-ARM-Prozessor. Um kompatible Instances zu finden, navigieren Sie zur EC2Amazon-Konsole. Wählen Sie Instance-Typen und suchen Sie dann nach`Architectures=arm64`.

AMIInhalt

- Unterstützter AWS Dienst: AWS PCS
- Betriebssystem: Amazon Linux 2
- Rechenarchitektur: arm64
- Linux-Kernel: 5.10.220-209.867.amzn2.aarch64
- EBSDatenträgertyp: gp2
- AWS PCSSlurm 23.11-Installationsprogramm: 23.11.9-1
- AWS PCSSoftwareinstallationsprogramm: 1.0.0-1
- EFAInstallationsprogramm: 1.33.0
- GDRCopy: 2.4
- NVIDIAFahrer: 535.154.05
- NVIDIACUDA: 12.2.2_535.104.05

Hinweise

- None

Veröffentlichungsdatum: 2024-08-22

Aktualisiert

- Keine. Erste Veröffentlichung.

Hinzugefügt

- Keine. Erste Veröffentlichung.

Entfernt

- Keine. Erste Veröffentlichung.

Dokumentverlauf für das AWS PCS-Benutzerhandbuch

In der folgenden Tabelle werden die Dokumentationsversionen für beschrieben AWS PCS.

Datum	Änderung	Aktualisierungen der Dokumentation	APIVersionen aktualisiert
28. August 2024	Seite „Verwaltete Richtlinien“ hinzugefügt	Weitere Informationen finden Sie unter AWS verwaltete Richtlinien für AWS Parallel Computing Service.	N/A
28. August 2024	AWS PCSVeröffentlichung	Erste Veröffentlichung 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.