使用者指南

AWS PCS

AWS PCS: 使用者指南

Table of Contents

什麼是 AWS 平行運算服務？

AWS 平行運算服務 (AWS PCS) 是一項受管服務，可讓您更輕鬆地執行和擴充高效能運算 (HPC) 工作負載，並在 AWS 使用 Slurm 上建置科學與工程模型。用 AWS PCS於建置整合最佳運 AWS 算、儲存、網路和視覺化的運算叢集。執行模擬或建立科學和工程模型。使用內建的管理和觀察功能，簡化叢集作業。讓使用者能夠在熟悉的環境中執行應用程式和工作，讓他們專注於研究和創新。

重要概念

中的叢集 AWS PCS具有 1 或多個佇列，與至少 1 個計算節點群組相關聯。工作會提交至佇列，並在運算節點群組定義的EC2執行個體上執行。您可以使用這些基礎來實作複雜的HPC架構。

叢集

叢集是用於管理資源和執行工作負載的資源。叢集是定義運算、網路、儲存體、身分識別和工作排程器組態組件的 AWS PCS資源。您可以透過指定要使用的工作排程器 (目前 Slurm)、您想要的排程器組態、要管理叢集的服務控制器，以及要在其VPC中啟動叢集資源來建立叢集。排程器接受並排程工作，並啟動處理這些工作的運算節點 (EC2執行個體)。

計算節點群組

計算節點群組是運算節點的集合，可 AWS PCS用來執行作業或提供叢集的互動式存取權。定義運算節點群組時，您可以指定常見特徵，例如 Amazon EC2 執行個體類型、最小和最大執行個體計數、目標VPC子網路、Amazon Machine Image (AMI)、購買選項和自訂啟動組態。 AWS PCS使用這些設定來有效率地啟動、管理和終止運算節點群組中的運算節點。

佇列

當您想要在特定叢集上執行工作時，請將它送出至特定佇列 (有時也稱為分割區)。工作會保留在佇列中，直到 AWS PCS排定它在計算節點群組上執行為止。您可以將一或多個計算節點群組與每個佇列產生關聯。使用工作排程器提供的各種排程原則，需要佇列才能在基礎計算節點群組資源上排程和執行工作。使用者不會將工作直接提交至計算節點或計算節點群組。

系統管理員

系統管理員部署、維護和操作叢集。他們可以 AWS PCS透過 AWS Management Console AWS PCSAPI、和存取 AWS SDK。他們可以通過SSH或訪問特定的集群 AWS Systems Manager，他們可以在那裡運行管理任務，運行作業，管理數據，並執行其他基於 shell 的活動。如需詳細資訊，請參閱 AWS Systems Manager 文件。

終端使用者

一般使用者沒有部署或操作叢集的 day-to-day 責任。他們使用終端機介面 (例如SSH) 來存取叢集資源、執行作業、管理資料，以及執行其他 shell 型活動。

設定 AWS 平行運算服務

完成下列工作以設定「 AWS 平行運算服務」(AWS PCS)。

主題

註冊一個 AWS 帳戶

如果您沒有 AWS 帳戶，請完成以下步驟來建立一個。

若要註冊成為 AWS 帳戶

1. 打開https://portal.aws.amazon.com/billing/註冊。
2. 請遵循線上指示進行。

   部分註冊程序需接收來電，並在電話鍵盤輸入驗證碼。

   當您註冊一個時 AWS 帳戶，將創建AWS 帳戶根使用者一個。根使用者有權存取該帳戶中的所有 AWS 服務 和資源。作為安全最佳實務，請將管理存取權指派給使用者，並且僅使用根使用者來執行需要根使用者存取權的任務。

AWS 註冊過程完成後，會向您發送確認電子郵件。您可以隨時前往 https://aws.amazon.com/並選擇「我的帳戶」，檢視目前的帳戶活動並管理您的帳戶。

建立具有管理存取權的使用者

註冊後，請保護您的 AWS 帳戶 AWS 帳戶根使用者 AWS IAM Identity Center、啟用和建立系統管理使用者，這樣您就不會將 root 使用者用於日常工作。

保護您的 AWS 帳戶根使用者

1. 選擇 Root 使用者並輸入您的 AWS 帳戶 電子郵件地址，以帳戶擁有者身分登入。AWS Management Console在下一頁中，輸入您的密碼。

如需使用根使用者登入的說明，請參閱 AWS 登入 使用者指南中的以根使用者身分登入。

2. 為您的 root 使用者開啟多因素驗證 (MFA)。

   如需指示，請參閱《使用指南》中的「IAM為 AWS 帳戶 root 使用者啟用虛擬MFA裝置 (主控台)」。

建立具有管理存取權的使用者

1. 啟用IAM身分識別中心。

   如需指示，請參閱 AWS IAM Identity Center 使用者指南中的啟用 AWS IAM Identity Center。

2. 在IAM身分識別中心中，將管理存取權授與使用者。

   若要取得有關使用 IAM Identity Center 目錄 做為身分識別來源的自學課程，請參閱《使用指南》IAM Identity Center 目錄中的「以預設值設定使用AWS IAM Identity Center 者存取」。

以具有管理存取權的使用者身分登入

- 若要使用您的 IAM Identity Center 使用者登入URL，請使用建立IAM身分識別中心使用者時傳送至您電子郵件地址的登入資訊。

  如需使用IAM身分識別中心使用者登入的說明，請參閱使用指南中的登入 AWS 存取入口網站。AWS 登入

指派存取權給其他使用者

1. 在 IAM Identity Center 中，建立遵循套用最低權限權限的最佳作法的權限集。

   如需指示，請參閱《AWS IAM Identity Center 使用者指南》中的建立許可集。

2. 將使用者指派至群組，然後對該群組指派單一登入存取權。

   如需指示，請參閱《AWS IAM Identity Center 使用者指南》中的新增群組。

安裝 AWS CLI

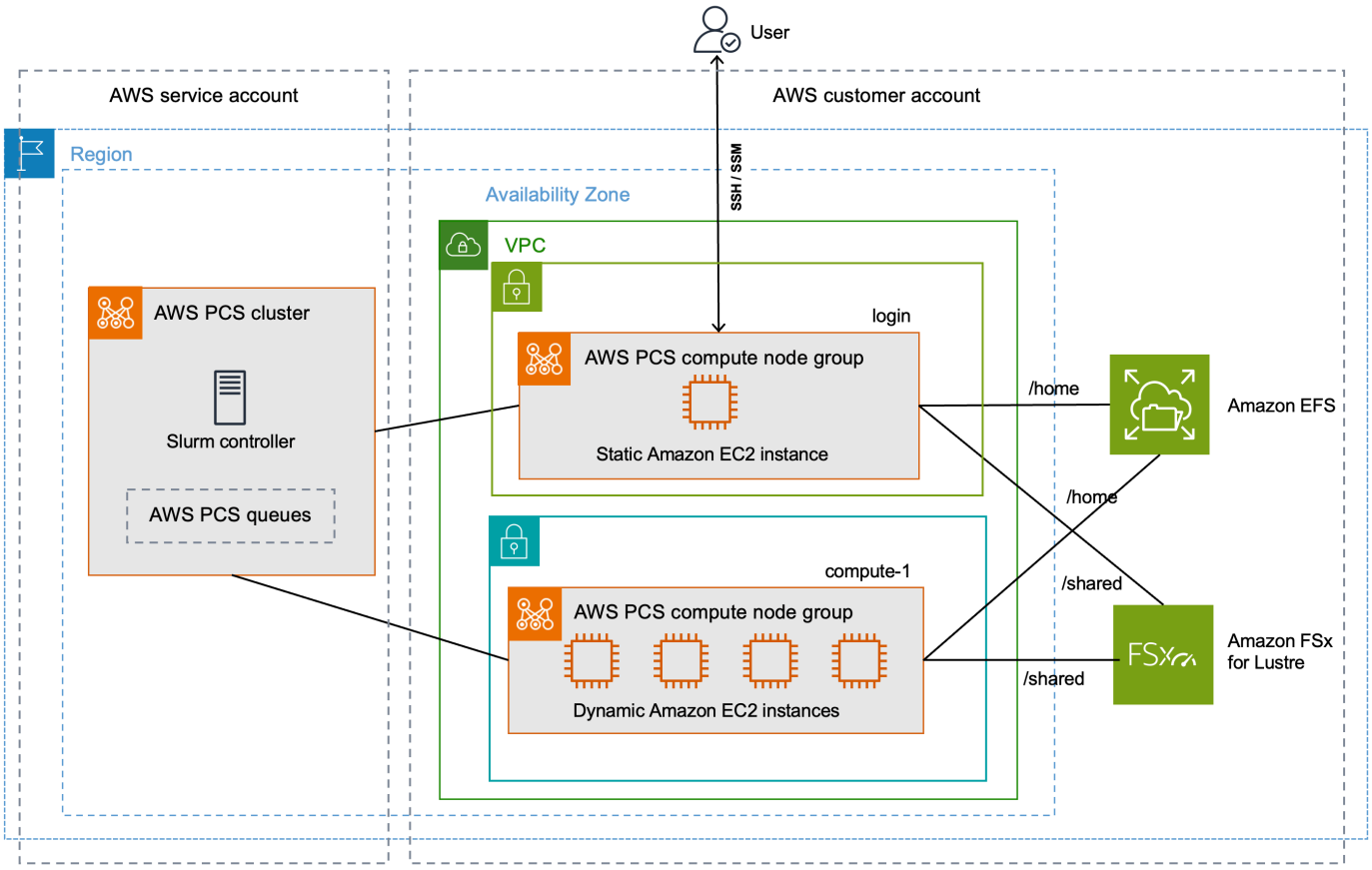
您必須使用最新版本的 AWS CLI. 如需相關資訊，請參閱第 2 版的AWS Command Line Interface 使用者指南 AWS CLI中的〈安裝或更新至最新版本〉。

在命令提示符下輸入以下命令以檢查您的 AWS CLI; 它應該顯示幫助信息。

```
aws pcs help
```

開始使用 AWS PCS

這是一個教程來創建一個簡單的集群，你可以用來嘗試 AWS PCS。下圖顯示了集群的設計。



本教程集群設計具有以下關鍵組成部分：

- 符合網[AWS PCS路需求的 A VPC 和子網路](#)。
- Amazon EFS 檔案系統，將用作共用主目錄。
- 一個 Amazon FSx 的 Lustre 文件系統，它提供了一個共享的高性能目錄。
- 一個 AWS PCS集群，它提供了一個 Slurm 控制器。
- 2 個運算節點群組。
  - `login`節點群組，提供對系統的殼層型互動式存取。
  - `compute-1`節點群組提供彈性調整規模的執行個體以執行作業。
- 1 個佇列，可將工作傳送至`compute-1`節點群組中的EC2執行個體。

叢集需要其他 AWS 資源，例如安全群組、IAM角色和EC2啟動範本，這些資源不會顯示在圖表中。

主題

開始使用的先決條件 AWS PCS

在開始本教學課程之前，請先安裝並設定下列建立和管理 AWS PCS叢集所需的工具和資源。

- AWS CLI— 用於處理 AWS 服務的命令行工具，包括 AWS PCS. 如需詳細資訊，請參閱《第 2 版 AWS Command Line Interface 使用者指南》 AWS CLI中的〈安裝或更新至最新版本〉。安裝之後 AWS CLI，我們建議您也對其進行配置。如需詳細資訊，請參閱第 2 版 AWS CLI的AWS Command Line Interface 使用者指南中的〈配置〉。
- 必要IAM權限 — 您使用的IAM安全性主體必須具有使用 AWS PCSIAM角色、服務連結角色 AWS CloudFormation VPC、和相關資源的權限。如需詳細資訊AWS 平行運算服務的 Identity and Access Management，請參閱《AWS Identity and Access Management 使用指南》中的「建立服務連結角色」。您必須以同一位使用者的身分完成本指南中的所有步驟。若要檢查目前使用者，請執行以下命令：

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

- 我們建議您在 Bash 殼層中完成本主題中的命令列步驟。如果您不使用 Bash shell，則某些指令碼命令 (如行接續字元以及設定和使用變數的方式) 需要針對 shell 進行調整。此外，您的 Shell 的引用及

轉義規則可能會有所不同。如需詳細資訊，請參閱第 2 版的《AWS Command Line Interface 使用者指南》中的〈〉中的〈含有字串的引號和具有字串的常值〉。 AWS CLI

建立以下項目的VPC和子網路 AWS PCS

您可以使用 CloudFormation 範本建立VPC和子網路。使用URL以下指令下載 CloudFormation 範本，然後在AWS CloudFormation 主控台中上傳範本以建立新 CloudFormation堆疊。如需詳細資訊，請參閱《使用指南》中的〈使AWS CloudFormation 用 AWS CloudFormation 主控台〉。

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

在 AWS CloudFormation 主控台中開啟範本的情況下，輸入下列選項。您可以使用範本中提供的預設值。

- 在 [提供堆疊名稱] 底下：
  - 在「堆疊名稱」下，輸入：

    ```
    hpc-networking
    ```
- 在參數之下：
  - 在下 VPC：
    - 在下 CidrBlock，輸入：

      ```
      10.3.0.0/16
      ```
  - 在子網路 A 之下：
    - 在 CidrPublicSubnetA 下，輸入：

      ```
      10.3.0.0/20
      ```
    - 在 CidrPrivateSubnetA 下，輸入：

      ```
      10.3.128.0/20
      ```
  - 在子網路 B 之下：
    - 在 CidrPublicSubnetB 下，輸入：

```
10.3.16.0/20
```

- 在 CidrPrivateSubnetB 下，輸入：

```
10.3.144.0/20
```

- 在子網路 C 下：
  - 對於 ProvisionSubnetsC，選擇真
  - 在 CidrPublicSubnetC 下，輸入：

```
10.3.32.0/20
```

  - 在 CidrPrivateSubnetC 下，輸入：

```
10.3.160.0/20
```

- 在功能之下：
  - 選中我確認 AWS CloudFormation 可能會創建IAM資源的複選框。

監視 CloudFormation 堆疊的狀態。當它到達時`CREATE_COMPLETE`，找到新的默認安全組的 ID VPC。您可以在稍後的自學課程中使用 ID。

尋找叢集的預設安全性群組 VPC

若要尋找新安全性群組中預設安全性群組的 IDVPC，請依照下列步驟執行：

- 導航到 Amazon VPC 控制台。
- 在「VPC儀表板」下，選取「篩選依據」VPC。
  - 選擇名稱VPC的開頭位置`hpc-networking`。
  - 在 [安全性] 下，選擇 [安全性群
- 尋找名為之群組的安全性群組 ID `default`。它有描述`default VPC security group`。稍後您可以使用 ID 來設定EC2啟動範本。

建立的安全性群組 AWS PCS

AWS PCS依賴安全群組來管理進出叢集及其計算節點群組的網路流量。如需此主題的詳細資訊，請參閱安全群組需求和考量事項。

在此步驟中，您將使用兩個安全性群組的 CloudFormation 範本。

- 叢集安全性群組，可在 AWS PCS控制器、計算節點和登入節點之間進行通訊。
- 輸入SSH安全性群組，您可以選擇性地將其新增至登入節點以支援SSH存取

建立的安全性群組 AWS PCS

您可以使用此 CloudFormation 範本建立VPC和子網路。使用URL以下指令下載 CloudFormation 範本，然後在AWS CloudFormation 主控台中上傳範本以建立新 CloudFormation 堆疊。如需詳細資訊，請參閱《使用指南》中的〈使AWS CloudFormation 用 AWS CloudFormation 主控台〉。

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

在 AWS CloudFormation 主控台中開啟範本的情況下，輸入下列選項。請注意，某些選項會預先填入範本中，您只需將它們保留為預設值即可。

- 在「提供堆疊名稱」下
  - 在堆疊名稱下，輸入：

    ```
    getstarted-sg
    ```
- 在參數之下
  - 在下 VpcId，選擇名稱VPC的開頭位置`hpc-networking`。
  - (選擇性) 在下 ClientIpCidr，為輸入SSH安全性群組輸入限制較嚴格的 IP 範圍。我們建議您使用您自己的 IP /子網路 (x.x.xx/32 代表您自己的 IP 或 x.x.x.x/24 的範圍限制此項目。 將 x.x.x.x 取代為您自己的 IP。PUBLIC 您可以使用諸如 https://ifconfig.co/ 之類的工具獲取您的公共 IP)

監視 CloudFormation 堆疊的狀態。當它到達`CREATE_COMPLETE`安全組資源已準備就緒。

建立了兩個安全群組，其名稱如下：

- `cluster-getstarted-sg`— 這是叢集安全性群組

- `inbound-ssh-getstarted-sg`— 這是一個允許入站SSH訪問的安全組

在中建立叢集 AWS PCS

在中 AWS PCS，叢集是用於管理資源和執行工作負載的永久性資源。您可以在新的或現有的子網路中為特定排程器 (AWS PCS目前支援 Slurm) 建立叢集。VPC叢集接受並排定工作，並啟動處理這些工作的運算節點 (EC2執行個體)。

若要建立叢集

1. 開啟AWS PCS主控台，然後選擇 [建立叢集]。
2. 在 [叢集設定] 區段中，輸入下列欄位：
   - 叢集名稱 — 輸入 `get-started`
   - 控制器大小 — 選擇小
3. 在 [網路功能] 區段中，選取下列欄位的值：
   - VPC— 選擇命VPC名 `hpc-networking:Large-Scale-HPC`
   - 子網路 — 選取名稱開頭為的子網路 `hpc-networking:PrivateSubnetA`
   - 安全群組 — 選取名為的叢集安全性群組 `cluster-getstarted-sg`
4. 選擇建立叢集。

> **Note**
>
> 狀態欄位會顯示在佈建叢集時建立。建立叢集可能需要幾分鐘的時間。

AWS PCS在 Amazon Elastic File System 中建立共用儲存

Amazon Elastic File System (AmazonEFS) 是一種提供無伺 AWS 服器、全彈性檔案儲存的服務，讓您無需佈建或管理儲存容量和效能即可共用檔案資料。如需詳細資訊，請參閱什麼是 Amazon Elastic File System？ 在 Amazon Elastic File System 使用者指南中。

示 AWS PCS範叢集使用EFS檔案系統在叢集節點之間提供共用的主目錄。在與叢集相同VPC的EFS檔案系統中建立檔案系統。

若要建立您的 Amazon EFS 檔案系統

1. 轉到 Amazon 控EFS制台。
2. 確保它設置為與您將嘗試相同的 AWS 區域 位置 AWS PCS。
3. 選擇 Create file system (建立檔案系統)。
4. 在 [建立檔案系統] 頁面上，設定下列參數：
   - 對於名稱，輸入 `getstarted-efs`。
   - 在虛擬私有雲（VPC）下，選擇VPC具名的 `hpc-networking:Large-Scale-HPC`
   - 選擇 Create (建立)。這會返回 [檔案系統] 頁面。
5. 記下檔案系統的檔**`getstarted-efs`**案系統 ID。您之後將會用到此資訊。

AWS PCS在 Amazon FSx 中為光澤創建共享存儲

Amazon FSx for Lustre 使啟動和運行流行的高性能 Lustre 文件系統變得輕鬆且經濟實惠。您可以將 Lustre 用於速度很重要的工作負載，例如機器學習、高效能運算 (HPC)、視訊處理和財務建模。有關更多信息，請參閱什麼是 Amazon FSx 的 Lustre？ 在 Amazon FSx 的光澤用戶指南。

示 AWS PCS範叢集可以使用 FSx for Lustre 檔案系統，在叢集節點之間提供高效能的共用目錄。在叢集中FSx為 Lustre 檔案系統建VPC立。

若要建立您FSx的 Lustre 檔案系統

1. 轉到 Amazon 控FSx制台。
2. 請確定主控台設定 AWS 區域 為與叢集相同。
3. 選擇 Create file system (建立檔案系統)。
   - 針對 [選取檔案系統類型]，選擇 Amazon FSx 做為 Lustre，然後選擇 [下一步]。
4. 在 [指定檔案系統詳細資訊] 頁面上，設定下列參數：
   - 在文件系統詳細信息
     - 對於名稱，輸入 `getstarted-fsx`。
     - 對於部署和儲存類型，請選擇「持續」，SSD
     - 對於每單位儲存體的輸送量，請選擇 125 MB /S/TIB
     - 對於儲存容量，請輸入 1.2 TiB

    - 在「中繼資料組態」中，選擇
    - 針對「資料壓縮類型」，選擇 LZ4
  - 在網絡和安全下
    - 對於虛擬私有雲 (VPC)，請選擇VPC具名的 `hpc-networking:Large-Scale-HPC`
    - 若為「VPC安全群組」，請將安全性群組保留為 `default`
    - 對於子網路，請選擇名稱開頭為的子網路 `hpc-networking:PrivateSubnetA`
  - 保留其他選項設定為其預設值。
  - 選擇 Next (下一步)。
5. 在 [檢閱並建立] 頁面上，選擇 [建立檔案系統]。這會返回 [檔案系統] 頁面。
6. 瀏覽至您所建立之 Lustre 檔案系統的詳細資訊頁面。FSx
7. 記下檔案系統 ID 和掛載名稱。您之後將會用到此資訊。

> **Note**
>
> 「狀態」欄位會在提供檔案系統時顯示「建立」。建立檔案系統可能需要幾分鐘的時間。請等到它完成，然後再繼續學習本教學課程的其餘部分。

在中建立計算節點群組 AWS PCS

運算節點群組是 AWS PCS啟動和管理之運算節點 (EC2執行個體) 的虛擬集合。定義運算節點群組時，您可以指定常見特徵，EC2例如執行個體類型、最小和最大執行個體計數、目標VPC子網路、偏好的購買選項和自訂啟動設定。 AWS PCS根據這些設定，可以啟動、管理和終止計算節點群組中的計算節點。示範叢集使用計算節點群組來提供使用者存取的登入節點，並使用個別的計算節點群組來處理作業。下列主題說明在叢集中設定這些計算節點群組的程序。

主題

- 建立執行個體設定檔 AWS PCS
- 建立啟動範本 AWS PCS
- 為登入節點建立計算節點群組 AWS PCS
- 建立運算節點群組以執行運算工作 AWS PCS

建立執行個體設定檔 AWS PCS

運算節點群組在建立時需要執行個體設定檔。如果您使用為 Amazon 建立角色EC2，主控台會自動建立執行個體設定檔，並為其指定與該角色相同的名稱。 AWS Management Console 如需詳細資訊，請參閱使用指南中的AWS Identity and Access Management 使用執行個體設定檔。

在下列程序中，您可以使用 AWS Management Console 為 Amazon 建立角色EC2，這也會為您的運算節點群組建立執行個體設定檔。

若要建立角色和執行個體設定檔

- 導覽至 IAM主控台。
- 在 Access management (存取管理) 下，請選擇 Policies (政策)。
  - 選擇 Create policy (建立政策)。
  - 在 [指定權限] 下，針對 [原則編輯器] 選擇JSON。
  - 以下列項目取代文字編輯器的內容：

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

  - 選擇 Next (下一步)。
  - 在 [檢閱並建立] 下，輸入做為 [原則名稱] `AWSPCS-getstarted-policy`。
  - 選擇 建立政策。
- 在 Access management (存取管理) 下，請選擇 Roles (角色)。
- 選擇建立角色。
- 在選取信任的實體之下：
  - 針對信任的實體類型，選取AWS 服務
  - 在使用案例之下，選擇EC2。

    - 然後，在 [選擇指定服務的使用案例] 下，選擇EC2。
  - 選擇 Next (下一步)。
- 在 [新增權限] 下：
  - 在 [權限] 原則中，搜尋 [開始] 原則AWSPCS。
  - 核取 AWSPCS-getStart-Started 原則旁邊的方塊，以將其新增至角色。
  - 在 [權限] 原則中，搜尋 A mazonSSMManaged InstanceCore。
  - 核取 A mazonSSMManaged InstanceCore 旁邊的核取方塊，將其新增至角色。
  - 選擇 Next (下一步)。
- 在「名稱」下，檢閱和建立：
  - 在角色詳細資料下：
    - 在角色名稱中，輸入 `AWSPCS-getstarted-role`。
  - 選擇建立角色。

建立啟動範本 AWS PCS

建立計算節點群組時，您會提供一個EC2啟動範本， AWS PCS用來設定其啟動的EC2執行個體。這包括安全群組和執行個體啟動時執行的指令碼等設定。

在此步驟中，將使用一個 CloudFormation 範本來建立兩個EC2啟動範本。其中一個範本將用於建立登入節點，另一個範本則用於建立運算節點。它們之間的主要區別在於可以將登錄節點配置為允許入站SSH訪問。

存取 CloudFormation 範本

使用URL以下指令下載 CloudFormation 範本，然後在AWS CloudFormation 主控台中上傳範本以建立新 CloudFormation 堆疊。如需詳細資訊，請參閱《使用指南》中的〈使AWS CloudFormation 用 AWS CloudFormation 主控台〉。

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

使用 CloudFormation 範本建立EC2啟動範本

使用下列程序在 AWS CloudFormation 主控台中完成 CloudFormation 範本

- 在 [提供堆疊名稱] 底下：

  - 在「堆疊名稱」下，輸入`getstarted-lt`。
- 在參數之下：
  - 在安全性下
    - 對於 VpcSecurityGroupId，選取叢集`default`中命名的安全性群組VPC。
    - 對於 ClusterSecurityGroupId，選取名為的群組 `cluster-getstarted-sg`
    - 對於 SshSecurityGroupId，選取名為的群組 `inbound-ssh-getstarted-sg`
    - 對於 SshKeyName，選取您偏好的 SSH key pair。
  - 在 [檔案系統]
    - 對於 EfsFilesystemId，輸入您先前在自學課程中建立的EFS檔案系統中的檔案系統 ID。
    - 對於 FSxLustreFilesystemId，輸入您先前在自學課程中建立FSx的 Lustre 檔案系統中的檔案系統 ID。
    - 對於 FSxLustreFilesystemMountName，請為 Lustre 檔案系統輸入相同FSx的掛載名稱。
- 選擇下一步，然後再次選擇下一步。
- 選擇提交。

監視 CloudFormation 堆疊的狀態。當它到達`CREATE_COMPLETE`啟動模板就可以使用了。

> **Note**
>
> 若要查看 CloudFormation 範本建立的所有資源，請開啟主AWS CloudFormation 控台。選擇 `getstarted-lt` 堆疊，然後選擇 Resources (資源) 索引標籤。

為登入節點建立計算節點群組 AWS PCS

運算節點群組是 AWS PCS啟動和管理之運算節點 (EC2執行個體) 的虛擬集合。定義運算節點群組時，您可以指定常見特徵，EC2例如執行個體類型、最小和最大執行個體計數、目標VPC子網路、偏好的購買選項和自訂啟動設定。 AWS PCS根據這些設定，可以啟動、管理和終止計算節點群組中的計算節點。

在此步驟中，您將啟動靜態運算節點群組，以提供叢集的互動式存取權。您可以使用SSH或 Amazon EC2 Systems Manager（SSM）登錄它，然後運行命令和管理 Slurm 任務。

若要建立計算節點群組

- 開啟主AWS PCS控台並瀏覽至 [叢集]。
- 選取名為的叢集 `get-started`
- 瀏覽至計算節點群組，然後選擇建立。
- 在 [計算節點群組設定] 區段中，提供下列資訊：
  - 計算節點群組名稱 — 輸入`login`。
- 在計算組態下，輸入或選取下列值：
  - EC2啟動範本 — 選擇名稱所在的啟動範本 `login-getstarted-lt`
  - IAM執行個體設定檔 — 選擇名為的執行個體 `AWSPCS-getstarted-role`
  - 子網路 — 選取名稱開頭為的子網路。`hpc-networking:PublicSubnetA`
  - 執行個體 — 選取`c6i.xlarge`。
  - 縮放組態 — 針對最小執行個體計數，輸入`1`。輸入做為執行個體數目上限`1`。
- 在其他設定下，指定下列項目：
  - AMIID — 選取名稱AMI的開頭位置 `aws-pcs-sample_ami-amzn2-x86_64-slurm-23.11`
- 選擇 [建立運算節點群組]。

「狀態」欄位會在佈建計算節點群組時顯示「建立」。您可以在自學課程中繼續進行下一個步驟。

建立運算節點群組以執行運算工作 AWS PCS

在此步驟中，您將啟動彈性擴展的計算節點群組，以執行提交至叢集的工作。

若要建立計算節點群組

- 開啟主AWS PCS控台並瀏覽至 [叢集]。
- 選取名為的叢集 `get-started`
- 瀏覽至計算節點群組，然後選擇建立。
- 在 [計算節點群組設定] 區段中，提供下列資訊：
  - 計算節點群組名稱 — 輸入`compute-1`。
- 在計算組態下，輸入或選取下列值：
  - EC2啟動範本 — 選擇名稱所在的啟動範本 `compute-getstarted-lt`
  - IAM執行個體設定檔 — 選擇名為的執行個體 `AWSPCS-getstarted-role`

  - 子網路 — 選取名稱開頭為的子網路。hpc-networking:PrivateSubnetA
  - 執行個體 — 選取c6i.xlarge。
  - 縮放組態 — 針對最小執行個體計數，輸入0。輸入做為執行個體數目上限4。
- 在其他設定下，指定下列項目：
  - AMIID — 選取名稱開頭的AMI位置aws-pcs-sample_ami-amzn2-x86_64-slurm-23.11。
- 選擇 [建立運算節點群組]。

「狀態」欄位會在佈建計算節點群組時顯示「建立」。

> **Important**
>
> 請等待「狀態」欄位顯示「作用中」，然後再繼續進行本教學課程的下一個步驟。

建立佇列以管理工作 AWS PCS

您可以將工作提交到佇列以執行它。工作會保留在佇列中，直到 AWS PCS排定它在計算節點群組上執行為止。每個佇列都與一或多個運算節點群組相關聯，這些群組提供EC2執行處理所需的執行個體。

在此步驟中，您將建立使用計算節點群組來處理作業的佇列。

建立佇列

- 開啟主AWS PCS控台。
- 選取名為的叢集get-started。
- 瀏覽至 [計算] 節點群組，並確定compute-1群組的狀態為 [作用中]。

> **Important**
>
> 在繼續下一個步驟之前，compute-1群組的狀態必須是「作用中」。

- 瀏覽至 [佇列] 並選擇 [建立佇列]。
  - 在佇列組態段落中，提供下列值：
    - 佇列名稱 — 輸入下列內容：demo
    - 計算節點群組 — 選取名為的計算節點群組compute-1。
- 選擇建立佇列。

建立佇列時，「狀態」欄位會顯示「建立中」。

> **Important**
>
> 請等待「狀態」欄位顯示「作用中」，然後再繼續進行本教學課程的下一個步驟。

Connect 到您的 AWS PCS叢集

`login`運算節點群組的狀態變為作用中之後，您就可以連線至其建立的EC2執行個體。

若要連線到登入節點

- 開啟主AWS PCS控台並瀏覽至 [叢集]。
- 選取名為的叢集`get-started`。
- 選擇 [計算節點群組]。
- 導覽至名為的計算節點群組`login`。
- 尋找計算節點群組識別碼。
- 在其他瀏覽器視窗或索引標籤中，開啟 Amazon EC2 主控台。
  - 選擇 Instances (執行個體)。
  - 搜尋具有下列標籤的EC2執行個體。Replace (取代) *`node-group-id`* 具有上一個步驟中計算節點群組識別碼的值。應該有 1 個實例。

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

  - Connect 至EC2執行個體。您可以使用工作階段管理員或SSH.

    Session Manager

    - 選取執行個體。
    - 選擇連線。
    - 在 Connect 至執行個體下，選取工作階段管理員
    - 選擇連線。
    - 選擇連線。互動式終端機會在您的瀏覽器中啟動。

    SSH

    - 選取執行個體。
    - 選擇連線。

- 在 Connect 至執行個體下，選取SSH用戶端。
- 依照主控台提供的指示操作。

> **Note**
>
> 執行個體的使用者名稱**ec2-user**不是root。

探索中的叢集環境 AWS PCS

登入叢集之後，您可以執行 shell 命令。例如，您可以變更使用者、在共用檔案系統上使用資料，以及與 Slurm 互動。

變更使用者

如果您已使用工作階段管理員登入叢集，您可能已連線為ssm-user。這是針對工作階段管理員建立的特殊使用者。使用以下命令切換到 Amazon Linux 2 上的默認用戶。如果您使用連接，則不需要執行此操作SSH。

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

使用共用檔案系統

您可以使用指令確認檔案EFS系統和 FSx Lustre 檔案系統是否可用。df -h叢集上的輸出應如下所示：

```
[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
```

/home文件系統掛載 127.0.0.1 並具有非常大的容量。這是您先前在自學課程中建立的EFS檔案系統。此處寫入的任何檔案都可/home在叢集中的所有節點下使用。

該/shared文件系統掛載一個私有 IP，並具有 1.2 TB 的容量。這是您先前在自學課程中建立的 Lustre 檔案系統。FSx此處寫入的任何檔案都可/shared在叢集中的所有節點下使用。

與思龍互動

主題

- 列出佇列和節點
- 顯示工作

列出佇列和節點

您可以列出佇列及其關聯使用的節點sinfo。叢集的輸出應如下所示：

```
[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 ~]$
```

請注意名為的分割區demo。它的狀態是up，它最多有 4 個節點。它與節點組中的compute-1節點相關聯。如果您編輯計算節點群組，並將執行個體數目上限增加到 8 個，則會讀取節點數目8並讀取節點清單compute-1-[1-8]。如果您建立第二個以 4 個節點命名test的運算節點群組，並將其新增至demo佇列，這些節點也會顯示在節點清單中。

顯示工作

您可以使用列出系統上任何狀態下的所有工作squeue。叢集的輸出應如下所示：

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

當您有 Slurm 工作擱置或執行中時，請稍後squeue再試一次執行。

執行單一節點工作 AWS PCS

若要使用 Slurm 執行工作，您可以準備指定工作需求的提交指令碼，並使用指令將其提交至佇列。sbatch通常，這是從共享目錄完成的，因此登錄和計算節點有一個共同的空間來訪問文件。

Connect 至叢集的登入節點，並在其 shell 提示字元中執行下列命令。

- 成為預設使用者。切換到共享目錄。

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

- 使用下列命令建立範例工作命令檔：

```
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
```

- 將工作指令碼提交至 Slurm 排程器：

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

- 提交作業後，它將返回一個作業 ID 作為一個數字。使用該 ID 來檢查工作狀態。Replace (取代) *job-id* 在下面的命令中，返回的數字sbatch。

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

Example

```
squeue --job 1
```

命squeue令會傳回類似下列內容的輸出：

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

- 繼續檢查工作的狀態，直到工作到達 R (執行中) 狀態為止。這項工作在squeue沒有返回任何東西時完成。

- 檢查/shared目錄的內容。

```
ls -alth /shared
```

命令輸出類似於以下內容：

```
-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
```

命名為single.1.out和的檔案single.1.err由叢集的其中一個計算節點寫入。由於工作是在共用目錄 (/shared) 中執行，因此您也可以在登入節點上使用它們。這就是為什麼您為此叢集設定 Lustre 檔案系統的原因。FSx

- 檢查single.1.out檔案的內容。

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

輸出類似以下內容：

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

使用 Slurm 執行多節點MPI作業 AWS PCS

這些說明演示了如何使用 Slurm 在中運行消息傳遞接口（MPI）作業。 AWS PCS

在登入節點的 shell 提示字元中執行下列命令。

- 成為預設使用者。切換到其主目錄。

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

- 在 C 編程語言中創建源代碼。

```
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
```

- 載入「開啟」MPI 模組。

```
module load openmpi
```

- 編譯 C 程序。

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

- 編寫 Slurm 工作提交腳本。

```
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
```

- 切換到共享目錄。

```
cd /shared
```

- 送出工作命令檔。

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

- 用squeue於監視工作，直到完成為止。
- 檢查以下內容multi.out：

```
cat multi.out
```

輸出類似如下。請注意，每個排名都有自己的 IP 地址，因為它在不同的節點上運行。

```
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
```

刪除您的 AWS 資源 AWS PCS

完成此教學課程所建立的叢集和節點群組之後，您應該刪除您所建立的資源。

> **Important**
>
> 您會針對在您的中執行的所有資源取得帳單費用 AWS 帳戶

刪除您為此教學課程建立的 AWS PCS資源的步驟

- 開啟主AWS PCS控台。
- 瀏覽至名為取得啟動的叢集。
- 導覽至 [佇列] 區段。
- 選取名為 demo 的佇列。
- 選擇 刪除 。

> **Important**
>
> 請等待佇列刪除後再繼續。

- 瀏覽至 [計算節點群組] 區段。
- 選取名為計算 -1 的計算節點群組。
- 選擇 刪除 。
- 選取名為 login 的計算節點群組。
- 選擇 刪除 。

> **Important**
> 請等待兩個計算節點群組都已刪除，然後再繼續。

- 在取得開始的叢集詳細資訊頁面中，選擇刪除。

> **Important**
> 請等待叢集已刪除，然後再繼續執行後續步驟。

刪除您為此教學課程建立的其他 AWS 資源的步驟

- 開啟主IAM控台。
  - 選擇角色。
  - 選取名為 AWSPCS-get 啟動角色的角色，然後選擇刪除。
  - 刪除角色之後，請選擇 [策略]。
  - 選取名為 AWSPCS-getStartted 原則的原則，然後選擇 [刪除]。
- 開啟 AWS CloudFormation 主控台。
  - 選取名為「開始-lt」的堆疊。
  - 選擇 刪除 。

> **Important**
> 在繼續之前，請等待堆疊刪除。

- 打開 Amazon EFS 控制台。
  - 選擇檔案系統。
  - 選取名為開始 EFS 的檔案系統。
  - 選擇 刪除 。

> **Important**
> 請等待檔案系統刪除，然後再繼續。

- 打開 Amazon FSx 控制台。

- 選擇檔案系統。
- 選擇名為「開始-fsx」的檔案系統。
- 選擇 刪除 。

> **Important**
>
> 請等待檔案系統刪除，然後再繼續。

- 開啟 AWS CloudFormation 主控台。
  - 選取名為「開始-sg」的堆疊。
  - 選擇 刪除 。
- 開啟 AWS CloudFormation 主控台。
  - 選取名為 hpc 網路的堆疊。
  - 選擇 Delete (刪除)。

使用 AWS PCS

本章提供協助您使用的資訊和指引 AWS PCS。

主題

AWS PCS叢集

AWS PCS叢集由下列元件組成：

- HPC系統排程器軟體的受管理執行個體，例如 Slurm 控制常駐程式 () `slurmctld`。
- 與HPC系統排程器整合以佈建和管理 Amazon EC2 執行個體的元件。
- 與HPC系統排程器整合以將日誌和指標傳輸到 Amazon 的元件 CloudWatch。

這些元件會在管理的帳戶中執行 AWS。他們共同合作來管理您客戶帳戶中的 Amazon EC2 執行個體。AWS PCS在 Amazon VPC 子網路中佈建彈性網路界面，以提供從排程器軟體到 Amazon EC2 執行個體的連線 (例如，支援排程批次任務，並讓使用者執行排程器命令以列出和管理這些任務)。

主題

在 AWS 並行計算服務中創建集群

本主題提供可用選項的概觀，並說明在「 AWS 平行運算服務」(AWS PCS) 中建立叢集時應考量的事項。如果這是您第一次建立 AWS PCS叢集，建議您遵循開始使用 AWS PCS。本教程可以幫助您創建一個工作HPC系統，而無需擴展到所有可能的可用選項和系統架構。

必要條件

- 符合AWS PCS网络需求VPC的現有子網路。在您部署叢集以供生產使用之前，建議您先全面了解VPC和子網路需求。若要建立VPC和子網路，請參閱VPC為您的 AWS PCS叢集建立。
- 具有建立及管理 AWS PCS資源之權限的IAM主參與者。如需詳細資訊，請參閱AWS 平行運算服務的 Identity and Access Management。

建立 AWS PCS叢集

您可以使用 AWS Management Console 或 AWS CLI 來建立叢集。

AWS Management Console

建立叢集

1. 在 https://console.aws.amazon.com/pcs/home#/ 叢集開啟 AWS PCS主控台，然後選擇 [建立叢集]。
2. 在 [叢集設定] 區段中，輸入下列欄位：
   - 叢集名稱 — 叢集的名稱。此名稱僅能使用英數字元 (區分大小寫) 和連字號。它必須以字母字元開頭，且長度不得超過 40 個字元。名稱在中必須是唯一的，而 AWS 區域 AWS 帳戶且您要在中建立叢集。
   - 排程器 — 選擇排程器和版本。 AWS PCS目前支持思盧姆 23.11。如需詳細資訊，請參閱思盧姆版本 AWS PCS。
   - 控制器大小 — 選擇控制器的大小。這決定了 AWS PCS叢集可以管理多少並行作業和計算節點。您只能在建立叢集時設定控制器大小。如需調整大小的詳細資訊，請參閱選擇 AWS PCS叢集大小。
3. 在「網路」段落中，選取下列欄位的值：
   - VPC— 選擇符合 AWS PCS要求VPC的現有. 如需詳細資訊，請參閱AWS PCSVPC以及子網路需求和考量。建立叢集之後，就無法變更叢集VPC。如果沒VPCs有列出，您必須先建立一個。

- 子網路 — VPC 會列出選取的所有可用子網路。在不同的可用區域中選擇兩個。每個子網路都必須符合 AWS PCS子網路需求。如需詳細資訊，請參閱AWS PCSVPC以及子網路需求和考量。我們建議您選取私有子網路，以避免將排程器端點暴露在公用網際網路中。
- 安全群組 — 指定您要與它為叢集建立 AWS PCS的網路介面關聯的安全性群組。您必須至少選取一個允許叢集及其計算節點之間通訊的安全性群組。如需詳細資訊，請參閱安全群組需求和考量事項。

4. (選擇性) 在「加密」(Encryption) 底下，您可以設定下列欄位來定義自訂金鑰來加密控制器資料：

   - KMS金鑰 ID — 保留為aws/pcs以使用PCS建立的KMS金鑰。選取現有的KMS金鑰別名以使用自訂KMS金鑰。請注意，用來建立叢集的帳戶必須擁有自訂KMS金鑰的kms:Decrypt權限。

5. （可選）在 Slurm 配置部分中，您可以指定 Slurm 配置選項，以覆蓋由以下方式設置的默認值： AWS PCS

   - 縮減閒置時間 — 這可控制動態佈建運算節點在放置在其上的工作完成或終止後，保持作用中的時間長度。將此值設定為較長的值可能會使後續工作更有可能在節點上執行，但可能導致成本增加。較短的值會降低成本，但可能會增加HPC系統佈建節點所花費的時間比例，而不是在節點上執行作業。
   - Prolog-這是計算節點組實例上 prolog 腳本目錄的完全合格路徑。這對應於思魯姆中的序言設置。請注意，這必須是一個目錄，而不是指向特定可執行文件的路徑。
   - Eilog — 這是運算節點群組執行個體上 epilog 指令碼目錄的完整路徑。這對應於 Slurm 中的「脫毛」設定。請注意，這必須是一個目錄，而不是指向特定可執行文件的路徑。
   - 選取類型參數 — 這有助於控制 Slurm 使用的資源選取演算法。將此值設定為CR_CPU_Memory將啟動記憶體感知排程，同時CR_CPU將其設定為CPU僅啟動排程。此參數對應於 Slurm 中的SelectTypeParameters設定，其中設定為 `select/cons_tres`「SelectType由」。 AWS PCS

6. (選擇性) 在標籤下，將任何標籤新增至 AWS PCS叢集。
7. 選擇建立叢集。建立叢集Creating時 AWS PCS會顯示 [狀態] 欄位。此程序需要幾分鐘的時間。

> **Important**
>
> 每 AWS 區域 個`Creating`狀態只能有 1 個叢集 AWS 帳戶。 AWS PCS當您嘗試建立叢集時，如果已經有叢集處於某個`Creating`狀態，則會傳回錯誤。

AWS CLI

建立叢集

1. 使用下列命令建立您的叢集。執行命令之前，請執行下列替換：

   - Replace (取代) *`region`* 使用您要在 AWS 區域 其中創建集群的 ID，例如`us-east-1`。
   - Replace (取代) *`my-cluster`* 使用叢集的名稱。此名稱僅能使用英數字元 (區分大小寫) 和連字號。它必須以字母字元開頭，且長度不得超過 40 個字元。名稱必須是唯一的 AWS 區域 ， AWS 帳戶 在您要建立叢集的位置。
   - Replace (取代) *`23.11`* 與任何受支持的思倫版本。

     > **Note**
     >
     > AWS PCS目前支持思盧姆 23.11。

   - Replace (取代) *`SMALL`* 任何受支援的叢集大小。這決定了 AWS PCS叢集可以管理多少並行作業和計算節點。它只能在建立叢集時進行設定。如需調整大小的詳細資訊，請參閱選擇 AWS PCS叢集大小。
   - `subnetIds`使用您自己的值取代的值。我們建議您選取私有子網路，以避免將排程器端點暴露在公用網際網路中。
   - 指定您`securityGroupIds`要與其 AWS PCS為叢集建立的網路介面關聯的介面。安全性群組必須與叢集位於VPC相同。您必須至少選取一個允許叢集及其計算節點之間通訊的安全性群組。如需詳細資訊，請參閱安全群組需求和考量事項。
   - 或者，您可以透過新增選項來微調 Slurm 行為。`--slurm-configration`例如，您可以使用將縮小閒置時間設定為 60 分鐘 (3600 秒)。`--slurm configuration scaleDownIdeTime=3600`
   - 或者，您可以使用提供自定義密KMS鑰來加密控制器的數據`--kms-key-id` *`kms-key`*。以現*`kms-key`*KMSARN有的金鑰 ID 或別名取代。請注意，用來建立叢集的帳戶必須擁有自訂KMS金鑰的`kms:Decrypt`權限。

```
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. 佈建叢集可能需要幾分鐘的時間。您可以使用下列命令來查詢叢集的狀態。在叢集的狀態欄位為之前，請勿繼續建立佇列或計算節點群組ACTIVE。

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

> **Important**
>
> 每 AWS 區域 個Creating狀態只能有 1 個叢集 AWS 帳戶。 AWS PCS當您嘗試建立叢集時，如果已經有叢集處於某個Creating狀態，則會傳回錯誤。

叢集的建議後續步驟

- 新增運算節點群組。
- 新增佇列。
- 啟用記錄。

刪除中的叢集 AWS PCS

本主題提供如何刪除AWSPCS叢集的概觀。

刪除 AWS PCS叢集時的考量

- 必須先刪除與叢集關聯的所有佇列，才能刪除叢集。如需詳細資訊，請參閱刪除佇列 AWS PCS。
- 必須先刪除與叢集關聯的所有計算節點群組，才能刪除叢集。如需詳細資訊，請參閱刪除中的計算節點群組 AWS PCS。

刪除叢集

您可以使用 AWS Management Console 或 AWS CLI 刪除叢集。

AWS Management Console

刪除叢集

1. 開啟主AWS PCS控台。
2. 選取要刪除的叢集。
3. 選擇 刪除 。
4. 叢集狀態欄位會顯示Deleting。此需要幾分鐘的時間來完成。

AWS CLI

刪除叢集

1. 使用下列命令刪除具有這些取代項目的叢集：
   - Replace (取代) *region-code* 與 AWS 區域 您的集群在。
   - Replace (取代) *my-cluster* 使用您的群集的名稱或 ID。

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

2. 刪除叢集可能需要幾分鐘的時間。您可以使用以下命令檢查叢集的狀態。

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

選擇 AWS PCS叢集大小

AWS PCS提供高可用性和安全性的叢集，同時自動執行修補、節點佈建和更新等關鍵工作。

建立叢集時，您可以根據兩個因素選取叢集的大小：

- 它將管理的運算節點數
- 您預期在叢集上執行的作用中佇列工作數目

泥漿叢集大小	受管理的執行個體數	作用中和排入佇列的工作數目
小型	最多可達 32 個	最多可達 256 個
中	最高可達 512	最多可達 8192
大型	最多可達	最多可達 16384

範例

- 如果您的叢集最多可有 24 個代管執行個體，且最多可執行 100 個工作，請選擇「小型」。
- 如果您的叢集最多可有 24 個代管執行個體，且最多可執行 1000 個工作，請選擇「中」。
- 如果您的叢集最多可有 1000 個代管執行個體，並且最多可執行 100 個工作，請選擇大型。
- 如果您的叢集最多可有 1000 個代管執行個體，且最多可執行 10,000 個工作，請選擇大型。

使用中的叢集密碼 AWS PCS

在建立叢集時， AWS PCS會建立叢集密碼，以連線至叢集上的工作排程器所需。您也可以建立 AWS PCS計算節點群組，這些群組定義要啟動的執行個體集，以回應擴展事件。 AWS PCS使用叢集密碼設定這些運算節點群組啟動的執行個體，以便它們可以連線到工作排程器。在某些情況下，您可能需要手動配置 Slurm 客戶端。範例包括建立持續性登入節點或設定具有工作管理功能的工作流程管理員。

AWS PCS將叢集密碼儲存為受管理的密碼，並pcs!在中加上前置詞 AWS Secrets Manager。秘密的費用包含在使用費用中 AWS PCS。

> **Warning**
>
> 請勿修改叢集密碼。 AWS PCS如果您修改叢集密碼，將無法與叢集通訊。 AWS PCS不支援叢集密碼的輪換。如果您需要修改叢集密碼，則必須建立新叢集。

內容

找到思盧姆集群秘密

您可以使用 AWS Secrets Manager 主控台或直接從 AWS PCS或API使用標籤尋找 AWS PCS受管理的密碼。

用 AWS Secrets Manager 來尋找叢集密碼

AWS Management Console

1. 導覽至「Sec rets Manager」主控台。
2. 選擇「密碼」，然後搜尋pcs!首碼。

> **Note**
>
> AWS PCS叢集密碼的名稱格pcs!slurm-secret-*cluster-id*式*cluster-id*為 AWS PCS叢集 ID。

AWS CLI

每個 AWS PCS叢集密碼也會以標記aws:pcs:*cluster-id*。您可以使用以下指令取得叢集的秘密 ID。執行命令之前，請先進行以下替換：

- *region*以取代 AWS 區域 以在中建立叢集，例如us-east-1。
- 以 AWS PCS叢集*cluster-id*的 ID 取代，以尋找其叢集密碼。

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

用 AWS PCS來尋找叢集密碼

您可以使用 AWS CLI 來尋找 AWS PCS叢集密碼。ARN輸入下列指令，進行下列取代：

- *region*以取代 AWS 區域 以在中建立叢集，例如us-east-1。
- 以叢集*my-cluster*的名稱或識別碼取代。

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

下面的示例輸出來自命get-cluster令。您可以使用secretArn並secretVersion一起獲得秘密。

```
{
    "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"
    }
}
```

獲取泥漿叢集的秘密

您可以使用 Secrets Manager 取得 Slurm 叢集密碼的目前 base64 編碼版本。下列範例使用. AWS CLI 執行命令之前，請先進行下列替換。

- *region*以取代 AWS 區域 以在中建立叢集，例如us-east-1。
- *secret-arn*以 AWS PCS叢集secretArn中的取代。

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

如需如何使用 Slurm 叢集密碼的相關資訊，請參閱。將獨立執行個體用作 AWS PCS登入節點

許可

您可以使用IAM主體來取得 Slurm 叢集密碼。IAM主體必須具有讀取密碼的權限。如需詳細資訊，請參閱《AWS Identity and Access Management 使用指南》中的「角色」術語和概念。

下列範例IAM原則允許存取叢集密碼範例。

```
{
    "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 PCS計算節點群組

AWS PCS運算節點群組是節點 (Amazon EC2 執行個體) 的邏輯集合。這些節點可用於執行運算作業，以及提供對HPC系統的互動式殼層型存取。運算節點群組包含建立節點的規則，包括要使用的

Amazon EC2 執行個體類型、要執行的執行個體數量、是否使用 Spot 執行個體還是隨需執行個體、要使用的子網路和安全群組，以及如何在啟動時設定每個執行個體。更新這些規則時， AWS PCS會更新與計算節點群組相關聯的資源以進行比對。

主題

在中建立計算節點群組 AWS PCS

本主題提供可用選項的概觀，並說明在「 AWS 平行運算服務」(AWS PCS) 中建立計算節點群組時應考量的事項。如果這是您第一次在中建立計算節點群組 AWS PCS，建議您遵循中的教學課程開始使用 AWS PCS。本教程可以幫助您創建一個工作HPC系統，而無需擴展到所有可能的可用選項和系統架構。

必要條件

- 足夠的服務配額可EC2在您的 AWS 區域. 您可以使用AWS Management Console來檢查並要求增加服務配額。
- 符合網路需求的現有子網路VPC和子 AWS PCS網路。建議您在部署叢集以供生產使用之前，先徹底瞭解這些需求。如需詳細資訊，請參閱AWS PCSVPC以及子網路需求和考量。您也可以使用 CloudFormation 範本來建立VPC和子網路。 AWS 提供 CloudFormation 模板的HPC配方。有關詳細資訊，請參閱中 aws-hpc-recipes的 GitHub。
- 具有呼叫 AWS PCS`RegisterComputeNodeGroupInstance`API動作和存取節點群組執行個IAM體所需任何其他 AWS 資源的執行個體設定檔。如需詳細資訊，請參閱IAM AWS 平行運算服務執行個體設定檔。
- 節點群組執行個體的啟動範本。如需詳細資訊，請參閱使用 Amazon EC2 啟動模板 AWS PCS。
- 若要建立使用 Amazon EC2 Spot 執行個體的運算節點群組，您 AWS 帳戶必須在. AWSServiceRoleForEC2Spot 如需詳細資訊，請參閱Amazon EC2 現貨角色 AWS PCS。

在以下位置建立計算節點群組 AWS PCS

您可以使用 AWS Management Console 或建立計算節點群組 AWS CLI。

AWS Management Console

使用主控台建立計算節點群組

1. 開啟主AWS PCS控台。
2. 選取要在其中建立計算節點群組的叢集。瀏覽至計算節點群組，然後選擇建立。
3. 在 [計算節點群組設定] 區段中，提供節點群組的名稱。名稱只能包含區分大小寫的英數字元和連字號。它必須以字母字元開頭，且長度不得超過 25 個字元。此名稱在叢集中必須是唯一的。
4. 在計算組態下，輸入或選取下列值：

   a. EC2啟動範本 — 選取要用於此節點群組的自訂啟動範本。Launch 範本可用來自訂網路設定，例如子網路和安全群組、監控組態和執行個體層級儲存體。如果您尚未準備啟動範本，請參閱使用 Amazon EC2 啟動模板 AWS PCS以瞭解如何建立啟動範本。

   > **Important**
   >
   > AWS PCS為每個計算節點群組建立受管理的啟動範本。這些都是命名的`pcs-identifier-do-not-delete`。建立或更新計算節點群組時，請勿選取這些群組，否則節點群組將無法正常運作。

   b. EC2啟動範本版本 — 選取自訂啟動範本的版本。您可以選擇一個特定的版本，這可以增強再現性。如果您稍後變更版本，則必須更新計算節點群組，以偵測啟動範本中的變更。如需詳細資訊，請參閱更新計 AWS PCS算節點群組。

   c. AMIID — 如果您的啟動範本不包含 AMI ID，或者您想要覆寫啟動範本中的值，請在此處提供 AMI ID。請注意，AMI用於節點群組的必須與相容 AWS PCS。您也可以選擇由 AMI提供的樣本 AWS。如需此主題的詳細資訊，請參閱Amazon 機器映像（AMIs） AWS PCS。

   d. IAM執行個體設定檔 — 選擇節點群組的執行個體設定檔。執行個體設定檔會授與執行個體安全存取 AWS 資源和服務的權限。如果您還沒有準備好，請參閱IAM AWS 平行運算服務執行個體設定檔以瞭解如何建立。

   e. 子網路 — 選擇 AWS PCS叢集部署VPC位置中的一或多個子網路。如果您選取多個子網路，節點之間的通EFA訊將無法使用，且不同子網路中節點之間的通訊可能會增加延遲。請確定您在此處指定的子網路符合您在EC2啟動範本中定義的任何子網路。

   f. 執行個體 — 選擇一或多個執行個體類型以滿足節點群組中的擴展請求。所有執行個體類型都必須具有相同的處理器架構 (x864_64 或 arm64) 和數目。vCPUs如果執行個體具有 GPUs，則所有執行個體類型的數目都必須相同GPUs。

g. 擴展配置 — 指定節點群組的執行個體數目下限和上限。您可以定義靜態配置 (節點正在執行的固定數目)，或定義動態組態 (最多可執行節點數目上限)。對於靜態配置，請將最小值和最大值設置為相同，大於零的數字。對於動態組態，請將最小執行個體設定為零，將最大執行個體設定為大於零的數字。 AWS PCS不支援混合使用靜態和動態執行個體的運算節點群組。

5. (選擇性) 在其他設定下，指定下列項目：

   a. 購買選項 — 選擇 Spot 執行個體和隨需執行個體。

   b. 配置策略 — 如果您已選取 Spot 購買選項，則可以指定在節點群組中啟動執行個體時如何選擇 Spot 容量集區。如需詳細資訊，請參閱 Amazon 彈性運算雲端使用者指南中的 Spot 執行個體配置策略。如果您已選取「隨需購買」選項，則此選項不會有任何作用。

6. (選擇性) 在「Slurm自訂設定」區段中，提供下列值：

   a. 重量 — 此值會設定群組中節點的優先順序，以供排程之用。權重較低的節點具有較高的優先級，並且單位是任意的。如需詳細資訊，請參閱Slurm文件中的重量。

   b. 實際記憶體 — 此值會設定節點群組中節點上的實際記憶體大小 (以 GB 為單位)。它旨在與中的叢集Slurm配置中的`CR_CPU_Memory`選項搭配使用 AWS PCS。如需詳細資訊，請參閱Slurm文件RealMemory中的。

7. (選用) 在標籤下，將任何標籤新增至您的計算節點群組。

8. 選擇 [建立運算節點群組]。[狀態] 欄位會`Creating`在 AWS PCS佈建節點群組時顯示。這可能需要幾分鐘的時間。

建議下一步

- 將節點群組新增至中的佇列， AWS PCS以使其能夠處理工作。

AWS CLI

若要使用建立您的計算節點群組 AWS CLI

使用以下指令建立佇列。執行命令之前，請執行下列替換：

1. Replace (取代) *`region`*具有要在中 AWS 區域 建立叢集的 ID，例如`us-east-1`。
2. Replace (取代) *`my-cluster`* 與您的群集`clusterId`的名稱或。

3. Replace (取代) *my-node-group*使用計算節點群組的名稱。此名稱僅能使用英數字元 (區分大小寫) 和連字號。它必須以字母字元開頭，且長度不得超過 25 個字元。此名稱在叢集中必須是唯一的。
4. Replace (取代) *subnet-ExampleID1* 包含叢集中的一或多個IDs子網VPC路。
5. Replace (取代) *lt-ExampleID1* 與您的自定義啟動模板的 ID。如果您還沒有準備好，請參閱使用 Amazon EC2 啟動模板 AWS PCS以瞭解如何建立。

   > **Important**
   >
   > AWS PCS為每個計算節點群組建立受管理的啟動範本。這些都是命名的pcs-*identifier*-do-not-delete。建立或更新計算節點群組時，請勿選取這些群組，否則節點群組將無法正常運作。

6. Replace (取代) *launch-template-version* 如果您要將節點群組與特定版本相關聯，請使用特定的啟動範本版本。
7. Replace (取代) *arn:InstanceProfile*使用您ARN的IAM實例配置文件。如果您沒有準備好，請參使用 Amazon EC2 啟動模板 AWS PCS閱指導。
8. Replace (取代) *min-instances* 以及 *max-instances* 具有整數值。您可以定義靜態配置 (節點正在執行的固定數目)，或定義動態組態 (最多可執行節點數目上限)。對於靜態配置，請將最小值和最大值設置為相同，大於零的數字。對於動態組態，請將最小執行個體設定為零，將最大執行個體設定為大於零的數字。 AWS PCS不支援混合使用靜態和動態執行個體的運算節點群組。
9. Replace (取代) *t3.large* 與另一個實例類型。您可以指定instanceType設定清單來新增更多執行個體類型。例如 *--instance-configs instanceType=c6i.16xlarge,instanceType=c6a.16xlarge*。 所有執行個體類型都必須具有相同的處理器架構 (x864_64 或 arm64) 和數目。vCPUs如果執行個體具有GPUs，則所有執行個體類型的數目都必須相同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
```

您可以將數個可選配置設置添加到命create-compute-node-group令中。

- 您可以指定自訂啟動範本是--amiId否不包含對的參考AMI，或是否要覆寫該值。請注意，AMI用於節點群組的必須與相容 AWS PCS。您也可以選擇由AMI提供的樣本 AWS。如需此主題的詳細資訊，請參閱Amazon 機器映像（AMIs）AWS PCS。
- 您可以使用在隨需 (ONDEMAND) 和 Spot (SPOT) 執行個體之間進行選擇--purchase-option。預設為「隨選」。如果您選擇 Spot 執行個體，您也可以使用--allocation-strategy來定義啟動節點群組中執行個體時如何 AWS PCS選擇 Spot 容量集區。如需詳細資訊，請參閱 Amazon 彈性運算雲端使用者指南中的 Spot 執行個體配置策略。
- 可以使用為節點群組中的節點提供組Slurm態選項--slurm-configuration。您可以設置權重（調度優先級）和實際內存。權重較低的節點具有較高的優先級，並且單位是任意的。如需詳細資訊，請參閱Slurm文件中的重量。實際記憶體會設定節點群組中節點上的實際記憶體大小 (以 GB 為單位)。它旨在與配置中的叢集CR_CPU_Memory選項搭Slurm配使用。 AWS PCS如需詳細資訊，請參閱Slurm文件RealMemory中的。

> **Important**
>
> 建立計算節點群組可能需要幾分鐘的時間。

您可以使用以下命令查詢節點組的狀態。在節點群組的狀態達到之前，您將無法將節點群組與佇列產生關聯ACTIVE。

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

更新計 AWS PCS算節點群組

本主題提供可用選項的概觀，並說明更新AWSPCS計算節點群組時應考量的事項。

更新計AWSPCS算節點群組的選項

更新AWSPCS計算節點群組可讓您變更由啟動之執行個體的屬性 AWSPCS，以及啟動這些執行個體的規則。例如，您可以將節點群組執行個體取代AMI為另一個安裝了不同軟體的執行個體。或者，您可以更新安全群組以變更輸入或輸出網路連線。您也可以變更擴展設定，甚至變更 Spot 執行個體的偏好購買選項。

下列節點群組設定在建立之後無法變更：

- 名稱
- 執行個體

更新 AWS PCS計算節點群組時的考量

計算節點群組會定義用於處理工作、提供互動式殼層存取權和其他工作的EC2執行個體。它們通常與一或多個 AWS PCS佇列相關聯。當您更新計算節點群組以變更其行為 (或其節點的行為) 時，請考慮下列事項：

- 當計算節點群組狀態從 [更新] 變更為 [使用中] 時，對計算節點群組屬性的變更會生效。使用更新的屬性啟動新執行個體。
- 不影響特定節點組態的更新不會影響執行中的節點。例如，新增子網路並變更配置策略。
- 如果您更新計算節點群組的啟動範本，則必須更新計算節點群組以使用新版本。
- 若要從計算節點群組中的節點新增或移除安全群組，請編輯其啟動範本並更新計算節點群組。使用更新的安全群組組啟動新執行個體。
- 如果您直接編輯計算節點群組所使用的安全性群組，則執行中和 future 執行個體會立即生效。
- 如果您在運算節點群組使用的IAM執行個體設定檔中新增或移除權限，則執行中和 future 執行個體會立即生效。
- 若要變更運算節點群組執行個體所使AMI用的執行個體，請更新計算節點群組 (或其啟動範本) 以使用新的執行個體，AMI然後等待 AWS PCS取代執行個體。
- AWS PCS在節點群組更新作業之後，取代節點群組中的現有執行個體。如果節點上有正在執行的工作，則可以在 AWS PCS取代節點之前完成這些工作。互動式使用者程序 (例如在登入節點執行個體上) 會終止。`Active`當 AWS PCS標示要取代的執行個體時，節點群組狀態會返回到，但實際的取代會在執行個體閒置時發生。
- 如果減少運算節點群組中允許的執行個體數目上限，請從 Slurm AWS PCS 移除節點以達到新的最大值。 AWS PCS終止與已移除的 Slurm 節點相關聯的執行中執行個體。已移除節點上的執行中工作會失敗，並返回其佇列。
- AWS PCS為每個計算節點群組建立受管理的啟動範本。它們被命名為`pcs-`*`identifier`*`-do-not-delete`。請勿在建立或更新計算節點群組時選取它們，否則節點群組將無法正常運作。
- 如果您更新計算節點群組以使用 Spot 進行購買選項，您的帳戶中必須具有AWSServiceRoleForEC2Spot服務連結角色。如需詳細資訊，請參閱Amazon EC2 現貨角色 AWS PCS。

更新AWSPCS計算節點群組

您可以使用AWS管理主控台或更新節點群組AWSCLI。

AWS Management Console

更新計算節點群組

1. 開啟AWSPCS主控台，位於：`https://console.aws.amazon.com/pcs/home#/clusters`
2. 選取您要更新計算節點群組的叢集。
3. 瀏覽至 [計算節點群組]，移至您要更新的節點群組，然後選取 [編輯]。
4. 在 [運算組態]、[其他設定] 和 [Slurm自訂設定] 區段中，更新以下任何值：
   - 執行個體 — 您無法變更運算節點群組中的執行個體。
5. 選擇更新。套用變更時，「狀態」欄位會顯示「更新」。

> **Important**
> 計算節點群組更新可能需要幾分鐘的時間。

AWS CLI

更新計算節點群組

1. 使用以下命令更新您的計算節點群組。執行命令之前，請執行下列替換：
   a. Replace (取代) *region-code* 與您要在其中創建集群的AWS區域。
   b. Replace (取代) *my-node-group* 使用名稱或`computeNodeGroupId`計算節點群組。
   c. Replace (取代) *my-cluster* 與您的群集`clusterId`的名稱或。

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

2. 更新除以外的任何節點群組參數`--instance-configs`。例如，要設置一個新的 AMI ID，傳遞到`--amiId my-custom-ami-id`哪裡 *my-custom-ami-id* 由您AMI的選擇取代。

> **Important**
>
> 更新計算節點群組可能需要幾分鐘的時間。

您可以使用以下命令查詢節點組的狀態。

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

刪除中的計算節點群組 AWS PCS

本主題提供可用選項的概觀，並說明在中刪除計算節點群組時應考量的事項 AWS PCS。

刪除計算節點群組時的考量

計算節點群組會定義用於處理工作、提供互動式殼層存取權和其他工作的EC2執行個體。它們通常與一或多個 AWS PCS佇列相關聯。刪除計算節點群組之前，請考慮下列事項：

- 由計算節點群組啟動的任何EC2執行個體都將終止。這將取消在這些執行個體上執行的工作，並終止執行中的互動式處理程序。
- 您必須先取消計算節點群組與所有佇列的關聯，然後才能刪除它。如需詳細資訊，請參閱更新 AWS PCS佇列。

刪除計算節點群組

您可以使用 AWS Management Console 或刪 AWS CLI 除計算節點群組。

AWS Management Console

若要刪除計算節點群組

1. 開啟主AWS PCS控台。
2. 選取計算節點群組的叢集。
3. 導覽至計算節點群組，然後選取要刪除的計算節點群組。
4. 選擇 刪除 。
5. 「狀態」字段會顯示`Deleting`。此需要幾分鐘的時間來完成。

> **Note**
>
> 您可以使用排程器原生的命令來確認已刪除計算節點群組。例如，使用sinfo或用squeue於泥漿。

AWS CLI

若要刪除計算節點群組

- 使用下列指令刪除運算節點群組，並取代這些項目：
  - Replace (取代) *region-code* 與 AWS 區域 您的集群在。
  - Replace (取代) *my-node-group* 使用計算節點群組的名稱或 ID。
  - Replace (取代) *my-cluster* 使用您的群集的名稱或 ID。

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

刪除計算節點群組可能需要幾分鐘的時間。

> **Note**
>
> 您可以使用排程器原生的命令來確認已刪除計算節點群組。例如，使用sinfo或用squeue於泥漿。

尋找運算節點群組執行個體 AWS PCS

每個 AWS PCS運算節點群組都可啟動具有共用設定的EC2執行個體 您可以使用EC2標籤在或中尋找運算節點群組中的執 AWS Management Console 行個體 AWS CLI。

AWS Management Console

尋找您的運算節點群組執行個體

1. 開啟主AWS PCS控台。

2. 選取 叢集。
3. 選擇計算節點群組。
4. 查找您創建的登錄節點組的 ID。
5. 導覽至EC2主控台並選擇 [執行個體]。
6. 使用下列標籤搜尋執行個體。Replace (取代) *node-group-id* 使用計算節點群組的 ID（不是名稱）。

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

7. (選擇性) 您可以在搜尋欄位中變更「執行處理」狀態的值，以尋找正在設定或最近終止的執行處理。
8. 在加標記的執行個體清單中找到每個執行個體的執行個體 ID 和 IP 位址。

AWS CLI

若要尋找節點群組執行個體，請使用下列指令。在執行指令之前，請進行下列取代作業：

- *region-code*以叢集 AWS 區域 的取代。範例：us-east-1
- *node-group-id*以計算節點群組的 ID (而非名稱) 取代。
- 以其他例證狀態 (例如pending或) 取running代，terminated以尋找其他狀態中的例EC2證。

```
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}
```

此命令會傳回如下輸出：的值PublicIP為，null如果執行個體位於私有子網路中。

```
[
    [
        {
            "InstanceID": "i-0123456789abcdefa",
            "State": "running",
            "PublicIP": "18.189.32.188",
```

```
            "PrivateIP": "10.0.0.1"
        }
    ]
]
```

> **Note**
>
> 如果您希describe-instances望傳回大量執行個體，則必須針對多個頁面使用選項。如需詳細資訊，請參閱 Amazon 彈性運算雲端API參考DescribeInstances中的。

使用 Amazon EC2 啟動模板 AWS PCS

在 Amazon 中EC2，啟動範本可以存放一組偏好設定，這樣您就不必在啟動執行個體時個別指定它們。 AWS PCS整合啟動範本，做為設定運算節點群組的靈活方式。建立節點群組時，您會提供啟動範本。 AWS PCS從中建立衍生的啟動範本，其中包含轉換，以協助確保其可與服務搭配使用。

瞭解撰寫自訂啟動範本時的選項和注意事項，可協助您撰寫要與搭配使用的範本 AWS PCS。如需有關啟動範本的詳細資訊，請參閱 Amazon EC2 使用者指南中的從啟動範本啟動執行個體從啟動範本啟動執行個體。

主題

概觀

您可以在EC2啟動範本中包含 30 多個參數，以控制執行個體設定方式的許多層面。大多數與完全兼容 AWS PCS，但也有一些例外。

EC2Launch 範本的下列參數會被忽略， AWS PCS因為這些屬性必須由服務直接管理：

- 實體類型/指定實體類型屬性 (**InstanceRequirements**) — AWS PCS 不支援以屬性為基礎的實體選取。

- 執行個體類型 (InstanceType) — 在建立節點群組時指定執行個體類型。
- 進@@ 階詳細資料/ IAM 執行個體設定檔 (IamInstanceProfile) — 您在建立或更新節點群組時提供此選項。
- 進@@ 階詳細資料/停用API終止 (DisableApiTermination) — AWS PCS 必須控制其啟動的節點群組執行個體的生命週期。
- 進@@ 階詳細資料/停用 API stop (DisableApiStop) — AWS PCS 必須控制其啟動的節點群組執行個體的生命週期。
- 進@@ 階詳細資料/停止 — 休眠行為 (HibernationOptions) — AWS PCS 不支援執行個體休眠。
- 進@@ 階細節/彈性 GPU (ElasticGpuSpecifications) — Amazon 彈性圖形於 2024 年 1 月 8 日結束使用壽命。
- 進@@ 階詳細資料/彈性推論 (ElasticInferenceAccelerators) — 新客戶無法再使用 Amazon Elastic Inference 論。
- AAdvanced詳細資料/指定每個核心CPU的選項/執行緒 (**ThreadsPerCore**) — 將每個核心的執行緒數 AWS PCS設定為 1。

這些參數具有支援以下相容性的特殊需求 AWS PCS：

- 使用者資料 (UserData) — 這必須是多部分編碼. 請參閱 使用 Amazon EC2 用戶數據。
- 應用程序和操作系統映像（ ImageId ）-你可以包括這個。但是，如果您在建立或更新節點群組時指定 AMI ID，它會覆寫啟動範本中的值。AMI您提供的必須相容 AWS PCS。如需詳細資訊，請參閱「Amazon 機器映像（ AMIs ） AWS PCS。
- 網@@ 路設定/防火牆 (安全群組) (SecurityGroups) — 無法在 AWS PCS啟動範本中設定安全群組名稱清單。您可以設定安全性群組 IDs (SecurityGroupIds) 清單，除非您在啟動範本中定義網路介面。然後，您必須IDs為每個介面指定安全性群組。如需詳細資訊，請參閱中的安全性群組 AWS PCS。
- 網@@ 路設定/進階網路設定 (NetworkInterfaces) — 如果您將EC2執行個體搭配單一網路卡使用，且不需要任何專門的網路組態，則 AWS PCS可以為您設定執行個體網路。若要在執行個體上設定多張網路卡或啟用彈性網狀架構介面卡，請使用NetworkInterfaces。每個網路介面都必須有IDs下的安全性群組清單Groups。如需詳細資訊，請參閱多個網絡接口 AWS PCS。
- 進@@ 階詳細資料/容量保留 (CapacityReservationSpecification) — 可以設定，但在使用CapacityReservationId時無法參考特定資料。 AWS PCS但是，您可以參考容量保留群組，其中該群組包含一或多個容量保留。如需詳細資訊，請參閱容量保留 AWS PCS。

建立基本的啟動範本

您可以使用 AWS Management Console 或建立啟動範本 AWS CLI。

AWS Management Console

建立啟動範本

1. 打開 Amazon EC2 控制台，然後選擇啟動模板。
2. 選擇 Create launch template (建立啟動範本)。
3. 在「啟動範本名稱與說明」下，為 Launch 範本名稱輸入唯一且獨特的名稱
4. 在 key pair 名稱的金鑰配對 (登入) 下，選取SSH要用來登入所管理EC2執行個體的金鑰配對 AWS PCS。此為選用操作，但建議您採用。
5. 在 [網路設定] 下方，然後選擇 [防火牆 (安全群組)] 下方的 [安全性群組] 以連結至網路介面 啟動範本中的所有安全性群組都必須來自您的 AWS PCS叢集VPC。至少選擇：
   - 允許與 AWS PCS叢集通訊的安全性群組
   - 安全性群組，可在由下列項目啟動的EC2執行個體間 AWS PCS
   - (選擇性) 允許互動式執行個體輸入SSH存取權的安全性群組
   - (選擇性) 允許計算節點向網際網路傳出連線的安全性群組
   - (選擇性) 允許存取網路資源 (例如共用檔案系統或資料庫伺服器) 的安全性群組。
6. 您可以在 Amazon EC2 主控台的「啟動範本」下存取新的啟動範本 ID。啟動範本 ID 將具有表單`lt-0123456789abcdef01`。

建議下一步

- 使用新的啟動範本建立或更新 AWS PCS計算節點群組。

AWS CLI

建立啟動範本

使用以下指令建立啟動範本。

- 執行命令之前，請執行下列替換：
  a. Replace (取代) *`region-code`* 與您正 AWS 區域 在使用的地方 AWS PCS

b. Replace (取代) *my-launch-template-name* 與您的模板的名稱。它對於您正在使用的 AWS 帳戶 和必 AWS 區域 須是唯一的。

c. Replace (取代) *my-ssh-key-name* 與您的首選SSH密鑰的名稱。

d. Replace (取代) *sg-ExampleID1* 以及 *sg-ExampleID2* 安全群組允許執行IDs個體與排程器之間的通訊，以及EC2執EC2行個體之間的通訊。如果您只有一個可啟用此流量的安全性群組，您可以移除sg-ExampleID2及其前面的逗號字元。您也可以新增更多安全性群組IDs。您在啟動範本中包含的所有安全性群組都必須來自 AWS PCS叢集VPC。

```
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"]}'
```

AWS CLI 將輸出類似下列的文字。您可以在中找到啟動範本 ID 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"
    }
}
```

建議下一步

- 使用新的啟動範本建立或更新 AWS PCS計算節點群組。

使用 Amazon EC2 用戶數據

您可以在執cloud-init行個體啟動時執行的啟動範本中提供EC2使用者資料。具有內容類型的使用者資料區塊會在cloud-config執行個體註冊前執行 AWS PCSAPI，而具有內容類型的使用者資料區塊會在註冊完成後text/x-shellscript執行，但在 Slurm 精靈啟動之前執行。如需內容類型的詳細資訊，請參閱 cloud-init 文件。

我們的用戶數據可以執行常見的配置方案，包括但不限於以下內容：

- 包括使用者或群組
- 安裝套件
- 建立分割區和檔案系統
- 掛載網路檔案系統

啟動範本中的使用者資料必須採用MIME多部分封存格式。這是因為您的使用者資料會與設定節點群組中的節點所需的其他 AWS PCS使用者資料合併。您可以將多個使用者資料圖塊合併為單一MIME多零件檔案。

MIME多零件檔案由下列元件組成：

- 內容類型和部分邊界宣告：`Content-Type: multipart/mixed; boundary="==BOUNDARY=="`
- MIME版本聲明：`MIME-Version: 1.0`
- 包含下列元件的一或多個使用者資料區塊：
  - 表示用戶數據塊的開始的開口邊界：`--==BOUNDARY==`。您必須將此邊界前的直線保持空白。
  - 塊的內容類型聲明：`Content-Type: text/cloud-config; charset="us-ascii"`或`Content-Type: text/x-shellscript; charset="us-ascii"`。您必須將內容類型宣告之後的行保留為空白。
  - 用戶數據的內容，例如 shell 命令或`cloud-config`指令列表。
- 表示MIME多零件檔案結尾的封閉邊界：`--==BOUNDARY==--`。您必須將封閉邊界前的直線保持空白。

> **Note**
>
> 如果您將使用者資料新增至 Amazon EC2 主控台中的啟動範本，則可以將其貼上為純文字。或者，您可以從文件上傳它。如果您使用 AWS CLI 或 AWS SDK，則必須先 base64 對使用者資料進行編碼，並在呼叫時將該字串作為`UserData`參數值提交 CreateLaunchTemplate，如此JSON檔案所示。

```
{
    "LaunchTemplateName": "base64-user-data",
```

```
    "LaunchTemplateData": {
        "UserData":
 "ewogICAgIkxhdW5jaFRlbXBsYXRlTmFtZSI6ICJpbmNyZWFzZS1jb250YWluZXItdm9sdW..."
    }
}
```

範例

- 範例：從套件儲存庫安裝軟體
- 範例：從 S3 儲存貯體執行指令碼
- 範例：設定全域環境變數
- 使用網路檔案系統 AWS PCS
- 範例：使用EFS檔案系統做為共用主目錄

範例： AWS PCS從套件儲存區域安裝軟體

提供此指令碼做為啟動範本"userData"中的值。如需詳細資訊，請參閱使用 Amazon EC2 用戶數據。

此指令碼使用 cloud config 在啟動時在節點群組執行個體上安裝軟體套件。如需詳細資訊，請參閱 cloud init 文件中的使用者資料格式。此範例會安裝curl和llvm。

> **Note**
>
> 您的執行個體必須能夠連線至其設定的套件儲存庫。

```
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==--
```

範例： AWS PCS從 S3 儲存貯體執行其他指令碼

提供此指令碼做為啟動範本"userData"中的值。如需詳細資訊，請參閱使用 Amazon EC2 用戶數據。

此指令碼使用 cloud config 從 S3 儲存貯體匯入指令碼，並在啟動時在節點群組執行個體上執行該指令碼。如需詳細資訊，請參閱 cloud init 文件中的使用者資料格式。

以您自己的詳細資料取代此指令碼中的下列值：

- *my-bucket-name* — 您的帳戶可以從中讀取的 S3 儲存貯體的名稱。
- *path* — 相對於 S3 儲存貯體根目錄的路徑。
- *shell* — 用來執行指令碼的 Linux 殼層，例如bash.

```
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==--
```

節點群組的IAM執行個體設定檔必須具有儲存貯體的存取權。下列IAM政策是上述使用者資料指令碼中儲存貯體的範例。

```
{
    "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/*"
```

```
            ]
        }
    ]
}
```

範例：設定全域環境變數 AWS PCS

提供此指令碼做為啟動範本"userData"中的值。如需詳細資訊，請參閱使用 Amazon EC2 用戶數據。

下列範例會使用/etc/profile.d在節點群組執行個體上設定全域變數。

```
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==--
```

範例：使用EFS檔案系統作為下列項目的共用主目錄 AWS PCS

提供此指令碼做為啟動範本"userData"中的值。如需詳細資訊，請參閱使用 Amazon EC2 用戶數據。

此範例擴充了EFS掛載範例，使用網路檔案系統 AWS PCS以實作共用主目錄。/home 的內容會在掛載EFS檔案系統之前進行備份。掛載完成後，內容會快速複製到共用儲存裝置上的位置。

以您自己的詳細資料取代此指令碼中的下列值：

- *`/mount-point-directory`* — 您要掛載EFS檔案系統之執行個體上的路徑。
- *`filesystem-id`* — 檔案系統的EFS檔案系統 ID。

```
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==--
```

啟用無密碼 SSH

您可以在共用主目錄範例上建置，以使用SSH金鑰在叢集執行個體之間實作SSH連線。對於每個使用共用主檔案系統的使用者，請執行類似下列內容的指令碼：

```
#!/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**
>
> 執行個體必須使用允許叢集節點之間SSH連線的安全性群組。

容量保留 AWS PCS

您可以使用隨需EC2容量保留或容量區塊在特定可用區域和特定持續時間內預留 Amazon EC2 容量，以確保在需要時具有必要的運算容量可用。

> **Note**
>
> AWS PCS支援隨需容量保留 (ODCR)，但目前不支援 ML 的容量區塊。

ODCRs搭配使用 AWS PCS

您可以選擇使 AWS PCS用預留執行個體的方式。如果您建立開啟的執行個體ODCR，您帳戶中 AWS PCS或其他程序啟動的任何相符執行個體都會計入保留項目。針對目標ODCR，僅針對保留項目啟動具有特定保留 ID 計數的執行個體。對於時間敏感的工作負載，ODCRs目標更為常見。

您可以將 AWS PCS計算節點群組新增至啟動範本，將其設定為使用目標ODCR。以下是執行此操作的步驟：

1. 建立目標隨需容量保留 (ODCR)。
2. 新增ODCR至容量保留群組。
3. 將「容量保留」群組與啟動範本產生關聯。
4. 建立或更新 AWS PCS計算節點群組以使用啟動範本。

範例：保留並使用具有目標性的 hpc6a.48xlarge 執行個體 ODCR

此範例指令會ODCR為 32 個 hpc6a.48xlarge 執行個體建立目標。若要啟動置放群組中的預留執行個體，`--placement-group-arn`請新增至指令。您可以使用和定義停止日期 `--end-date``--end-date-type`，否則保留將繼續進行，直到手動終止為止。

```
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
```

此命令的結果將是新ARN的ODCR。若要ODCR與搭配使用 AWS PCS，必須將其新增至容量保留群組。這是因為 AWS PCS不支持個人ODCRs。如需詳細資訊，請參閱 Amazon 彈性運算雲端使用者指南中的容量保留群組。

以下說明如何將名為的容量保留群組新增ODCR至容量保留群組`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
```

透過ODCR建立並新增至容量保留群組，現在可以透過將其新增至啟動範本來連接至 AWS PCS計算節點群組。以下是參照容量保留群組的啟動範本範例。

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

最後，建立或更新 AWS PCS運算節點群組以使用 hpc6a.48xlarge 執行個體，並使用參考其容量保留群組中的啟動範本。ODCR對於靜態節點群組，請將執行個體下限和上限設定為保留項目的大小 (32)。對於動態節點群組，請將執行個體下限設定為 0，並將最大執行個體設定為保留大小。

此範例是針對一個計算節點群組佈建ODCR的單一簡單實作。但是， AWS PCS支持許多其他設計。例如，您可以在多個計算節點群組之間細分大型ODCR或容量保留群組。或者，您可以使用ODCRs已創建並與您共享的另一個AWS帳戶。索引鍵限制是必須一ODCRs律包含在容量保留群組中。

如需詳細資訊，請參閱 Amazon 彈性運算雲端使用者指南中的 ML 適用的隨需容量保留和容量區塊。

實用的啟動範本參數

本節說明一些可能廣泛有用的啟動範本參數。 AWS PCS

開啟詳細 CloudWatch監控

您可以使用啟動樣板參數，以較短的間隔啟用 CloudWatch 測量結果收集。

AWS Management Console

在用於建立或編輯啟動範本的主控台頁面上，此選項位於 [進階詳細資料] 區段下。將詳細 CloudWatch 監控設定為啟用。

YAML

```
Monitoring:
    Enabled: True
```

JSON

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

如需詳細資訊，請參閱 Amazon 彈性運算雲端 Linux 執行個體使用者指南中的執行個體啟用或關閉詳細監控。

執行處理中繼資料服務版本 2 (IMDSv2)

搭配EC2執行個體使用 IMDS v2 可提供重大的安全性增強功能，並協助降低在 AWS 環境中存取執行個體中繼資料時

AWS Management Console

在用於建立或編輯啟動範本的主控台頁面上，此選項位於 [進階詳細資料] 區段下。將可存取的中繼資料設定為 [已啟用]、[中繼資料版本] 為 [僅 V2] (需要 Token)，將中繼資料回應躍點

YAML

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

JSON

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

.

AWS PCS佇列

AWS PCS佇列是排程器原生工作佇列實作的輕量級抽象。在 Slurm 的情況下， AWS PCS隊列相當於 Slurm 分區。

使用者會將工作提交至它們所在的佇列，直到排定為在一或多個計算節點群組所提供的節點上執行為止。一個 AWS PCS叢集可以有多個工作佇列。例如，您可以建立將 Amazon EC2 隨需執行個體用於高優先順序任務的佇列，以及使用 Amazon EC2 Spot 執行個體執行低優先順序任務的另一個佇列。

主題

在中建立佇列 AWS PCS

本主題提供可用選項的概觀，並說明在中建立佇列時應考量的事項 AWS PCS。

必要條件

- AWSPCS叢集-佇列只能與特定PCS叢集建立關聯。
- 一或多個AWSPCS計算節點群組-佇列必須與至少一個PCS計算節點群組相關聯。

若要在中建立佇列 AWS PCS

您可以使用 AWS Management Console 或建立佇列 AWS CLI。

AWS Management Console

使用主控台建立佇列

1. 開啟AWSPCS主控台，位於：`https://console.aws.amazon.com/pcs/home#/clusters`
2. 選取要在其中建立佇列的叢集。瀏覽至 [佇列] 並選擇 [建立佇列]。
3. 在佇列組態段落中，提供下列值：
   a. 佇列名稱 — 佇列的名稱。此名稱僅能使用英數字元 (區分大小寫) 和連字號。它必須以字母字元開頭，且長度不得超過 25 個字元。名稱在叢集中必須是唯一的。
   b. 計算節點群組 — 選取一或多個計算節點群組來為此佇列提供服務。一個計算節點群組可以與多個佇列相關聯。
4. (選擇性) 在「標籤」下，將任何標籤新增至AWSPCS佇列

5. 選擇建立佇列。設定佇列時，[狀態] 欄位會顯示 [正在建立]。建立佇列可能需要幾分鐘的時間。

建議下一步

- 將工作提交到新佇列

AWS CLI

若要使用建立佇列 AWS CLI

使用以下指令建立佇列。執行命令之前，請執行下列替換：

1. Replace (取代) *region-code* 與您要在其中創建集群的AWS區域。
2. Replace (取代) *my-queue* 與您的隊列的名稱。此名稱僅能使用英數字元 (區分大小寫) 和連字號。它必須以字母字元開頭，且長度不得超過 25 個字元。名稱在叢集中必須是唯一的。
3. Replace (取代) *my-cluster* 與您的群集 clusterId 的名稱或。
4. `computeNodeGroupId`以您自己的計算節點群組識別碼取代的值。請注意，您無法在建立佇列時指定計算節點群組名稱。

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

建立佇列可能需要幾分鐘的時間。您可以使用以下命令查詢隊列的狀態。在工作狀態達到之前，您將無法提交工作至佇列`ACTIVE`。

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

建議下一步

- 將工作提交到新佇列

更新 AWS PCS佇列

本主題提供可用選項的概觀，並說明更新AWSPCS佇列時應考量的事項。

更新AWSPCS佇列時的考量

佇列更新不會影響執行中的工作，但叢集可能無法在佇列更新時接受新的工作。

更新AWSPCS計算節點群組

您可以使用AWS管理主控台或更新節點群組AWSCLI。

AWS Management Console

更新佇列

1. 開啟AWSPCS主控台的位置：`https://console.aws.amazon.com/pcs/home#/clusters`
2. 選取您要更新佇列的叢集。
3. 瀏覽至 [佇列]，移至想要更新的佇列，然後選取 [編輯]。
4. 在佇列組態區段中，更新下列任一值：
   - 節點群組 — 新增或移除與佇列關聯的計算節點群組。
   - 標籤 — 新增或移除佇列的標籤。
5. 選擇更新。套用變更時，「狀態」欄位會顯示「更新」。

> **Important**
> 佇列更新可能需要幾分鐘的時間。

AWS CLI

更新佇列

1. 使用下列指令更新佇列。執行命令之前，請執行下列替換：
   a. Replace (取代) *region-code* 使用 AWS 區域 您想要在其中創建集群的.
   b. Replace (取代) *my-queue* 使用名稱或您computeNodeGroupId的隊列。

c. Replace (取代) *my-cluster* 與您的群集clusterId的名稱或。

d. 若要變更計算節點群組關聯，請提供的更新清單--compute-node-group-configurations。

- 例如，若要新增第二個計算節點群組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. 更新佇列可能需要幾分鐘的時間。您可以使用以下命令查詢隊列的狀態。在工作狀態達到之前，您將無法提交工作至佇列ACTIVE。

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

建議的後續步驟

- 將工作提交至您更新的佇列。

刪除佇列 AWS PCS

本主題提供如何在中刪除佇列的概觀 AWS PCS。

刪除佇列時的考量

- 如果佇列中有工作正在執行，則排程器會在刪除佇列時終止這些工作。佇列中的擱置工作將會被取消。考慮等待佇列中的工作完成，或者使用排程器的原生命令 (例如 scancel Slurm) 手動停止/取消它們。

刪除佇列

您可以使用 AWS Management Console 或 AWS CLI 刪除佇列。

AWS Management Console

刪除佇列

1. 開啟主AWS PCS控台。
2. 選取佇列的叢集。
3. 導覽至 [佇列]，然後選取要刪除的佇列。
4. 選擇 刪除 。
5. 「狀態」字段會顯示`Deleting`。此需要幾分鐘的時間來完成。

> **Note**
>
> 您可以使用排程器原生的指令來確認佇列已刪除。例如，使用`sinfo`或用`squeue`於泥漿。

AWS CLI

刪除佇列

- 使用下列指令來刪除佇列，並使用這些取代物：
  - Replace (取代) *region-code* 與 AWS 區域 您的集群在。
  - Replace (取代) *my-queue* 使用隊列的名稱或 ID。
  - Replace (取代) *my-cluster* 使用您的群集的名稱或 ID。

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

刪除佇列可能需要幾分鐘的時間。

> **Note**
>
> 您可以使用排程器原生的指令來確認佇列已刪除。例如，使用`sinfo`或用`squeue`於泥漿。

AWS PCS登入節點

AWS PCS叢集通常需要至少 1 個登入節點來支援互動式存取和工作管理。若要達成此目的，其中一個方法是設定為登入節點功能的靜態 AWS PCS運算節點群組。您也可以設定獨立EC2執行個體做為登入節點。

主題

使用 AWS PCS計算節點群組提供登入節點

本主題提供建議組態選項的概觀，並說明當您使用AWSPCS計算節點群組來提供叢集持續、互動式存取時應考量的事項。

建立登入節點的 AWS PCS計算節點群組

在操作上，這與建立一般運算節點群組沒有太大不同。但是，有一些關鍵的配置選擇：

- 為計算節點群組中至少一個EC2執行個體設定靜態擴展配置。
- 選擇隨需購買選項，以避免回收您的執行個體。
- 選擇計算節點群組的資訊性名稱，例如登入。
- 如果您希望登入節點執行個體可在您的外部存取VPC，請考慮使用公有子網路。
- 如果您打算允許SSH存取，啟動範本將需要一個安全性群組，將SSH連接埠公開給您選擇的 IP 位址。
- IAM執行個體設定檔應該只有您希望使用者擁有的AWS權限。如需詳細資訊，請參閱 IAM AWS 平行運算服務執行個體設定檔。
- 考慮讓 AWS Systems Manager 工作階段管理員管理您的登入執行個體
- 考慮將執行個體AWS認證的存取權限限制為只有管理使用者

- 選取比一般運算節點群組便宜的執行個體類型，因為登入節點將持續執行。
- 使用與其他計算節點群組相同的 (或衍生產品)AMI，以協助確保所有執行個體都安裝了相同的軟體。若要取得有關自訂的更多資訊，請參閱〈AMIs Amazon 機器映像（AMIs） AWS PCS
- 在您的登入節點上設定相同的網路檔案系統 (Amazon EFS、Amazon FSx to Lustre 等)，就像在您的運算執行個體上一樣。如需詳細資訊，請參閱使用網路檔案系統 AWS PCS。

存取您的登入節點

新的計算節點群組達到ACTIVE狀態後，您可以找到它建立的EC2執行個體並登入它們。如需詳細資訊，請參閱尋找運算節點群組執行個體 AWS PCS。

更新登入節點的 AWS PCS計算節點群組

您可以使用更新登入節點群組 UpdateComputeNodeGroup。作為節點群組更新程序的一部分，執行中的執行個體將會被取代。請注意，這會中斷執行個體上的任何作用中使用者工作階段或程序。執行中或排入佇列的 Slurm 工作將不受影響。如需詳細資訊，請參閱更新計 AWS PCS算節點群組。

您也可以編輯計算節點群組使用的啟動範本。您必須使用 UpdateComputeNodeGroup 將更新的啟動範本套用至計算節點群組。在計算節點群組中啟動的新EC2執行個體會使用更新的啟動範本。如需詳細資訊，請參閱使用 Amazon EC2 啟動模板 AWS PCS。

刪除登入節點的 AWS PCS計算節點群組

您可以使用中的刪除計算節點群組機制來更新登入節點群組 AWS PCS。執行中的執行個體會在刪除節點群組的過程中終止。請注意，這會中斷執行個體上的任何作用中使用者工作階段或程序。執行中或排入佇列的 Slurm 工作將不受影響。如需詳細資訊，請參閱刪除中的計算節點群組 AWS PCS。

將獨立執行個體用作 AWS PCS登入節點

您可以設定獨立EC2執行個體，以與 AWS PCS叢集的 Slurm 排程器互動。這對於建立使用 AWS PCS叢集但在管理之外運作的登入節點、工作站或專用工作流程 AWS PCS管理主機非常有用。若要這麼做，每個獨立執行個體都必須：

1. 已安裝相容的 Slurm 軟體版本。
2. 能夠連線至 AWS PCS叢集的 Slurmctld 端點。
3. 使用 AWS PCS叢集的端點和密碼正確配置 Slurm 驗證和 Cred Kiosk 守護程序 (sackd)。如需詳細資訊，請參閱 Slurm 文件中的 sackd。

本教學課程可協助您設定連線至 AWS PCS叢集的獨立執行個體。

內容

步驟 1 — 擷取目標 AWS PCS叢集的位址和密碼

使用下列命令擷取有關目標 AWS PCS叢集 AWS CLI 的詳細資訊。執行命令之前，請執行下列替換：

- Replace (取代) *region-code* 與目標叢集執行的 AWS 區域 位置。
- Replace (取代) *cluster-ident* 包含目標叢集的名稱或識別碼

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

該命令將返回類似於此示例的輸出。

```
{
    "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"
        }
    }
}
```

在此範例中，叢集 Slurm 控制器端點具有的 IP 位址，`10.3.149.220`並在連接埠上執行。`6817``secretArn`將在稍後的步驟中使用來擷取叢集密碼。IP 位址和連接埠將在稍後的步驟中用來設定`sackd`服務。

步驟 2 — 啟動EC2執行個體

啟動 EC2 執行個體

1. 打開 Amazon EC2 控制台。
2. 在導覽窗格中，選擇 Instances (執行個體)，接著選擇 Launch Instances (啟動執行個體) 來開啟新的啟動執行個體精靈。
3. (選擇性) 在 [名稱和標籤] 區段中，提供執行個體的名稱，例如`PCS-LoginNode`。該名稱將指派作為執行個體的資源標籤 (`Name=PCS-LoginNode`)。
4. 在「應用程式和作業系統映像」區段中，AMI為支援的其中一個作業系統選取一個 AWS PCS。如需詳細資訊，請參閱支援的作業系統。
5. 在「例證類型」區段中，選取支援的例證類型。如需詳細資訊，請參閱支援的執行個體類型。
6. 在 [key pair] 區段中，選取SSH要用於執行個體的金鑰配對。
7. 在「網路設定」區段中：
    - 選擇編輯。

i. 選取 AWS PCS叢集VPC的。

ii. 對於「防火牆（安全群組）」，選擇 「選取現有安全群組」。

A. 選取允許執行個體和目標 AWS PCS叢集 Slurm 控制器之間流量的安全性群組。如需詳細資訊，請參閱安全群組需求和考量事項。

B. (選擇性) 選取允許執行個體輸入SSH存取權的安全性群組。

8. 在儲存區段中，視需要設定儲存磁碟區。請務必設定足夠的空間來安裝應用程式和程式庫，以啟用您的使用案例。

9. 在 [進階] 下，選擇允許存取叢集密碼的IAM角色。如需詳細資訊，請參閱獲取泥漿叢集的秘密。

10. 在「摘要」窗格中，選擇 Launch 執行個體。

第 3 步-在實例上安裝思盧姆

當執行個體啟動並變為使用中時，請使用您偏好的機制連線至執行個體。使用提供的 Slurm 安裝程式，將 Slurm 安裝 AWS 到執行個體上。如需詳細資訊，請參閱思盧姆安裝程序。

下載 Slurm 安裝程序，將其解壓縮，然後使用`installer.sh`腳本安裝 Slurm。如需詳細資訊，請參閱步驟 3 — 安裝思隆。

步驟 4 — 擷取並儲存叢集密碼

這些指示需要 AWS CLI. 如需詳細資訊，請參閱《第 2 版AWS Command Line Interface 使用者指南》AWS CLI中的〈安裝或更新至最新版本〉。

使用下列命令儲存叢集密碼。

- 建立 Slurm 的組態目錄。

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

- 擷取、解碼和儲存叢集密碼。在運行此命令之前，請替換 *region-code* 與目標群集運行的區域，並替換 *secret-arn* 使用在步驟 1 中`secretArn`擷取的值。

```
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**
>
> 在多使用者環境中，任何具有執行個體存取權的使用者都可以存取執行個體中繼資料服務 (IMDS)，就可以擷取叢集密碼。反過來，這可能允許他們模擬其他用戶。請考慮限制只有 root 使用IMDS者或系統管理使用者的存取權。或者，考慮使用不依賴於實例配置文件來獲取和配置密鑰的不同機制。

- 設置 Slurm 密鑰文件的所有權和權限。

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

> **Note**
>
> Slurm 金鑰必須是sackd服務執行身分的使用者和群組所擁有。

步驟 5 — 設定 AWS PCS叢集的連線

若要建立 AWS PCS叢集的連線，請依照下列步驟以系統服務的sackd身分啟動。

1. 使用下列指令設定sackd服務的環境檔案。在運行命令之前，請替換 *ip-address* 以及 *port* 使用在步驟 1 中從端點擷取的值。

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

2. 建立用於管理sackd程序的systemd服務檔案。

```
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. 設定sackd服務檔案的所有權。

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

4. 啟用sackd服務。

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

5. 啟動 sackd 服務。

```
sudo systemctl start sackd
```

步驟 6 — (選擇性) 測試連線

確認sackd服務正在執行。範例輸出如下。如果有錯誤，它們通常會顯示在這裡。

```
[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
```

使用 Slurm 用戶端命令 (例如和)，確認與叢集的連線正在運作`sinfo`。`squeue`這裡是從示例輸出`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]
```

您也應該能夠提交工作。例如，類似此範例的命令會在叢集中的 1 個節點上啟動互動式工作。

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

AWS PCS网络

您的 AWS PCS群集是在 Amazon 中創建的VPC。本章包含下列有關叢集排程器和節點之網路功能的主題。

除了選擇要在其中啟動執行個體的子網路外，您必須使用EC2啟動範本為 AWS PCS計算節點群組設定網路。如需啟動範本的詳細資訊，請參閱使用 Amazon EC2 啟動模板 AWS PCS。

主題

- AWS PCSVPC以及子網路需求和考量
- VPC為您的 AWS PCS叢集建立
- 中的安全性群組 AWS PCS
- 多個網絡接口 AWS PCS
- EC2例證的放置群組 AWS PCS
- 使用彈性織物適配器（EFA） AWS PCS

AWS PCSVPC以及子網路需求和考量

當您建立 AWS PCS叢集時，您可以在其中指定VPC一個子網路VPC。本主題提供您與叢集搭配使用之VPC和子網路的 AWS PCS特定需求和考量事項的概觀。如果您沒有與一VPC起使用 AWS PCS，則可

以使用 AWS提供的模板創建一個 AWS CloudFormation 模板。如需詳細資訊VPCs，請參閱 Amazon VPC 使用者指南中的虛擬私有雲端 (VPC)。

VPC需求和考量

建立叢集時，您指定的VPC叢集必須符合下列需求和考量：

- VPC必須有足夠數量的 IP 位址可供叢集、任何節點以及您要建立的其他叢集資源使用。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的您VPCs和子網路的 IP 定址。
- VPC必須具有DNS主機名稱和DNS解析度支援。否則，節點無法註冊客戶叢集。如需詳細資訊，請參閱 Amazon VPC 使用者指南VPC中的DNS屬性。
- VPC可能需要VPC端點 AWS PrivateLink 使用才能連絡 AWS PCSAPI. 如需詳細資訊，請參閱 Amazon 使用VPC者指南 AWS PrivateLink中的使用 Connect VPC 到服務。

子網需求和注意事項

當您建立 Slurm 叢集時， AWS PCS會在您指定的子網路中建立彈性網路介面 (ENI)。此網路介面可讓排程器控制器與客戶之間進行通訊VPC。網路介面也可讓 Slurm 與客戶帳戶中部署的元件進行通訊。您只能在建立時指定叢集的子網路。

叢集的子網路要求

您在建立叢集時指定的子網路必須符合下列需求：

- 子網路必須至少有 1 個 IP 位址供使用 AWS PCS。
- 子網路不能位於 AWS Outposts AWS Wavelength、或 AWS 本機區域中。
- 子網路可以是公用或私有子網路。如果可能，建議您指定私有子網路。公用子網路是具有路由表的子網路，其中包含通往網際網路閘道的路由；私有子網路是具有路由表的子網路，不包含通往網際網路閘道的路由。

節點的子網路要求

您可以將節點和其他叢集資源部署到您在建立 AWS PCS叢集時指定的子網路，以及其他相同VPC的子網路。

您部署節點和叢集資源的任何子網路必須符合下列需求：

- 您必須確定子網路具有足夠的可用 IP 位址來部署所有節點和叢集資源。
- 如果您計劃將節點部署到公用子網路，則該子網路必須自動指派IPv4公用位址。

- 如果您將節點部署到的子網路是私有子網路，且其路由表不包含網路位址轉譯 (NAT) 裝置 (IPv4) 的路由，請使用 AWS PrivateLink 給客戶新增VPC端點VPC。VPC節點所接觸的所有 AWS 服務都需要端點。唯一需要的端點是 AWS PCS允許節點調用該`registerNodeGroupInstances`API操作。
- 公用或私有子網路狀態不會影響 AWS PCS；必須可連線所需的端點。

VPC為您的 AWS PCS叢集建立

您可以在 AWS 平行運算服務 (VPC) 中為叢集建立 Amazon Virtual Private Cloud (Amazon AWS PCS)。

使用 Amazon VPC 將VPC資源啟動到您定義的虛擬網路中。此虛擬網路非常近似於您在自有資料中心內運作的傳統網路。但是，它帶來了使用 Amazon Web 服務的可擴展基礎架構的好處。建議您在部署生產VPC叢集之前，先徹底瞭解 Amazon VPC 服務。如需詳細資訊，請參閱什麼是 AmazonVPC？ 在作者視覺模式下。 Amazon VPC 用戶指南。

PCS叢集、節點和支援資源 (例如檔案系統和目錄服務) 都會部署在 Amazon 中VPC。如果您想要VPC搭配使用現有的 AmazonPCS，它必須符合中所述的要求AWS PCSVPC以及子網路需求和考量。本主題說明如何使用 AWS—提供VPC的範 AWS CloudFormation 本建立符合PCS需求的。部署範本後，您可以檢視範本所建立的資源，以確切了解其建立的資源以及這些資源的組態。

必要條件

要VPC為創建一個 AmazonPCS，您必須具有創建 Amazon VPC 資源的必要IAM許可。這些VPCs資源包括子網路、安全群組、路由表和路由，以及網際網路和NAT閘道。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的使用公VPC有子網路建立。若要查看 Amazon 的完整清單EC2，請參閱服務授權參考EC2中的 Amazon 動作、資源和條件金鑰。

創建一個 Amazon VPC

VPC通過複製創建一個並粘貼URL適合您將使用的 AWS 區域 位置PCS。您也可以下載 AWS CloudFormation 模板並自己上傳到AWS CloudFormation 控制台。

- US East (N. Virginia) (美國東部 (維吉尼亞北部)) (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
```

- US East (Ohio) (美國東部 (俄亥俄)) (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
```

- US West (Oregon) (美國西部 (奧勒岡)) (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
```

- 僅限範本

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

要創建一個 Amazon VPC PCS

1. 在AWS CloudFormation 主控台中開啟範本。

   > **Note**
   >
   > 這些會預先填入範本中，以便您可以將它們保留為預設值。

2. 在 [提供堆疊名稱] 下，然後輸入 [堆疊名稱] `hpc-networking`。
3. 在參數下，輸入下列詳細資訊：

   a. 在下 VPC，然後 CidrBlock，輸入 `10.3.0.0/16`

   b. 在子網路 A 之下：

      i. 然後 CidrPublicSubnetA，輸入 `10.3.0.0/20`

      ii. 然後 CidrPrivateSubnetA，輸入 `10.3.128.0/20`

   c. 在子網路 B 之下：

      i. 然後 CidrPublicSubnetB，輸入 `10.3.16.0/20`

      ii. 然後 CidrPrivateSubnetA，輸入 `10.3.144.0/20`

   d. 在子網路 C 下：

      i. 對於 ProvisionSubnetsC，請選取`True`。

> **Note**
>
> 如果您要VPC在具有少於三個可用區域的區域中建立，則將忽略此選項如果設定為True。

ii. 然後 CidrPublicSubnetB，輸入 10.3.32.0/20

iii. 然後 CidrPrivateSubnetA，輸入 10.3.160.0/20

4. 在 [功能] 下，核取 [我確認AWS CloudFormation 可能會建立IAM資源] 方塊。

監視 AWS CloudFormation 堆疊的狀態。當它到達時CREATE_COMPLETE，資VPC源已準備好供您使用。

> **Note**
>
> 若要查看 AWS CloudFormation 範本建立的所有資源，請開啟主AWS CloudFormation 控台。選擇 hpc-networking 堆疊，然後選擇 Resources (資源) 索引標籤。

中的安全性群組 AWS PCS

Amazon 中的安全群組EC2充當虛擬防火牆，以控制執行個體的輸入和輸出流量。使用 AWS PCS計算節點群組的啟動範本，在其執行個體中新增或移除安全群組。如果您的啟動範本不包含任何網路介面，請SecurityGroupIds使用提供安全性群組清單。如果您的啟動範本定義了網路介面，您必須使用Groups參數將安全性群組指派給每個網路介面。如需啟動範本的詳細資訊，請參閱使用 Amazon EC2 啟動模板 AWS PCS。

> **Note**
>
> 啟動範本中的安全群組組態變更只會影響計算節點群組更新後啟動的新執行個體。

安全群組需求和考量事項

AWS PCS在您建立叢集時指定的子網路中建立跨帳戶彈性網路介面 (ENI)。這會提供HPC排程器 (在所管理的帳戶中執行) 與由 AWS啟動的EC2執行個體通訊的路徑 AWS PCS。您必須提供一個安全群組，以允ENI許排程器ENI與叢集EC2執行個體之間進行雙向通訊。

若要達成此目的，就是建立一個寬容自我參照安全群組，該群組允許群組所有成員之間所有連接埠上的 TCP /IP 流量。您可以將其附加到叢集和節點群組EC2執行個體。

寬容安全群組組態範例

規則類型	通訊協定	連接埠	來源	目的地
傳入	全部	全部	自我	
傳出	全部	全部		0.0.0.0/0
傳出	全部	全部		自我

這些規則允許所有流量在 Slurm 控制器和節點之間自由流動，允許所有輸出流量到任何目的地，並啟用EFA流量。

限制性安全群組組態範例

您也可以限制叢集及其運算節點之間的開啟連接埠。對於 Slurm 排程器，連接到叢集的安全性群組必須允許下列連接埠：

- 6817 — 啟用`slurmctld`來EC2自執行個體的輸入連線
- 6818 — 啟用從執行個體`slurmd`執行`slurmctld`到執行個體的輸出連線 EC2

連接到計算節點的安全性群組必須允許下列連接埠：

- 6817 — 啟用`slurmctld`來自EC2執行個體的輸出連線。
- 6818 — 啟用節點群組執行個體`slurmd`的傳入`slurmctld`和輸出連線 `slurmd`
- 60001—63000 — 節點群組執行個體之間的輸入和輸出連線以支援 `srun`
- EFA節點群組執行個體間的流量。如需詳細資訊，請參閱 Linux 執行個體使用者指南中的準備EFA已啟用的安全性群組。
- 工作負載所需的任何其他節點間流量

多個網絡接口 AWS PCS

有些EC2執行個體有多張網路卡。這使得它們能夠提供更高的網路效能，包括 100 Gbps 以上的頻寬能力，以及改善的封包處理能力。如需有關具有多張網路卡的執行個體的詳細資訊，請參閱 Amazon 彈性運算雲端使用者指南中的彈性網路界面。

透過將網路介面新增至其EC2啟動範本，為 AWS PCS運算節點群組中的執行個體設定其他網路卡。以下是啟用兩張網路卡的範例啟動範本，例如可在hpc7a.96xlarge執行個體上找到。請注意下列詳細資訊：

- 每個網路介面的子網路必須與您在設定將使用啟動範本的 AWS PCS計算節點群組時所選擇的子網路相同。
- 透過設定 a 來建立主要網路裝置 (例如SSH和HTTPS流量) DeviceIndex 的例行網路通訊0。其他網路介面具DeviceIndex有1. 只能有一個主要網路介面 — 所有其他介面都是次要介面。
- 所有網路介面都必須具有唯一的NetworkCardIndex。建議的做法是按照啟動範本中定義的順序對其進行編號。
- 每個網路介面的安全性群組均使用設定Groups。在此範例中，輸入SSH安全性群組 (sg-*SshSecurityGroupId*) 會新增至主要網路介面，以及啟用叢集內通訊的安全性群組 (sg-*ClusterSecurityGroupId*)。最後，允許輸出連線到網際網路 (sg-*InternetOutboundSecurityGroupId*) 的安全性群組會新增至主要和次要介面。

```
{
    "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"]
        }
```

```
    ]
}
```

EC2例證的放置群組 AWS PCS

您可以使用放置群組來影響EC2執行個體的放置，以符合執行個體的工作負載需求。

放置群組類型

- 叢集 — 封裝在可用區域中靠近的執行個體，以針對低延遲通訊進行最佳化。
- 磁碟分割 — 將執行個體分散到邏輯分割區，以協助最大化
- 傳播 — 嚴格強制執行少量執行個體在不同的硬體上啟動，這也有助於恢復。

如需詳細資訊，請參閱 Amazon 彈性運算雲端使用者指南中的 Amazon EC2 執行個體放置群組。

當您將 AWS PCS計算節點群組設定為使用彈性網狀架構介面卡 (EFA) 時，建議您加入叢集放置群組。

若要建立可搭配使用的叢集置放群組 EFA

1. 為計算節點群組建立具有叢集類型的置放群組。
   - 使用以下 AWS CLI 命令：

     ```
     aws ec2 create-placement-group --strategy cluster --group-name PLACEMENT-GROUP-NAME
     ```

   - 您也可以使用 CloudFormation 範本來建立放置群組。若要取得更多資訊，請參閱《使用指南》中的〈AWS CloudFormation 使用 CloudFormation樣板〉。從下面下載模板URL並將其上傳到CloudFormation 控制台。

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

2. 在 AWS PCS計算節點群組的EC2啟動範本中包括放置群組。

使用彈性織物適配器（EFA） AWS PCS

Elastic Fabric Adapter (EFA) 是一種高效能的進階網路互連，您 AWS 可以將其連接至EC2執行個體，以加速高效能運算 (HPC) 和機器學習應用程式。啟用在 AWS PCS叢集上執行的應用程式包EFA括將 AWS PCS計算節點群組執行個體設定EFA為使用，如下所示。

內容

安裝EFA在兼 AWS PCS容 AMI

AWS PCS計算節點群組中AMI使用的EFA驅動程式必須安裝並載入。如需有關如何使用已安裝EFA軟體建置自訂AMI的資訊，請參閱自定義 Amazon 機器映像（AMIs）AWS PCS。

識別EFA啟用EC2執行個

若要使用EFA， AWS PCS計算群組允許的所有執行個體類型都必須支援EFA，且必須具有相同數量的vCPUs (GPUs如果適用)。如需EFA已啟用執行個體的清單，請參閱 Amazon 彈性運算雲端使用者指南EC2中的 Amazon 上適用HPC和 ML 工作負載的彈性網狀架構適配器。您也可以使用 AWS CLI 檢視支援的執行個體類型清單EFA。Replace (取代) *region-code* 與您使用的 AWS 區域 地方 AWS PCS，例如`us-east-1`。

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

判斷有多少可用的網路介面

有些EC2執行個體有多張網路卡。這允許他們有多個EFAs。如需詳細資訊，請參閱多個網絡接口 AWS PCS。

建立安全性群組以支援EFA通訊

AWS CLI

您可以使用下列 AWS CLI 命令來建立支援的安全性群組EFA。此指令會輸出安全性群組 ID。進行以下替換：

- *region-code*— 指定 AWS 區域 您使用的位置 AWS PCS，例如us-east-1。
- *vpc-id*— 指定您使用的 ID AWS PCS。VPC
- *efa-group-name*— 為安全性群組提供您選擇的名稱。

```
aws ec2 create-security-group \
    --group-name efa-group-name \
    --description "Security group to enable EFA traffic" \
    --vpc-id vpc-id \
    --region region-code
```

使用下列命令來附加輸入和輸出安全性群組規則。進行以下替換：

- *efa-secgroup-id*— 提供您剛剛創建的EFA安全組的 ID。

```
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

您可以使用 CloudFormation 範本建立支援的安全性群組EFA。從下列項目下載範本URL，然後將範本上傳至主AWS CloudFormation 控台。

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

在 AWS CloudFormation 主控台中開啟範本的情況下，輸入下列選項。

- 在「提供堆疊名稱」下
  - 在「堆疊名稱」下，輸入名稱，例如`efa-sg-stack`。
- 在參數之下
  - 在下 SecurityGroupName，輸入名稱，例如`efa-sg`。
  - 在下 VPC，選取您VPC要使用的位置 AWS PCS。

完成 CloudFormation 堆疊的建立並監視其狀態。當它到達EFA安全組已準備就緒可`CREATE_COMPLETE`以使用。

(選擇性) 建立放置群組

建議啟動叢集置放群組EFA中使用的所有執行個體，以將它們之間的實體距離降到最低。我們建議您為要使用的每個計算節點群組建立放置群組EFA。請參閱EC2例證的放置群組 AWS PCS若要為您的計算節點群組建立置放群組。

建立或更新EC2啟動範本

EFA網路介面是在 AWS PCS運算節點群組的EC2啟動範本中設定的。如果有多個網路卡，則EFAs可以設定多個網路卡。EFA安全性群組和選用的放置群組也包含在啟動範本中。

以下是具有兩張網路卡之執行個體的啟動範例，例如 hpc 7a.96xlarge。執行個體將會在叢集置放群組`subnet-`*`SubnetID1`*中啟動`pg-`*`PlacementGroupId1`*。

安全性群組必須特別新增至每個EFA介面。每個人都EFA需要啟用EFA流量 (`sg-`*`EfaSecGroupId`*) 的安全性群組。其他安全性群組，尤其是處理常規流量 (例如SSH或HTTPS) 的安全性群組只需要附加至主要網路介面 (由 a `DeviceIndex` 的指定`0`)。已定義網路介面的啟動範本不支援使用`SecurityGroupIds`參數設定安全群組 — 您必須在設定的每個網路介面`Groups`中設定值。

```
{
    "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"]
        }
    ]
}
```

建立或更新計算節點群組

使用具有相同數量、相同處理器架構且全部支援的執行個體vCPUs，建立或更新 AWS PCS運算節點群組EFA。設定計算節點群組，使其AMI與安裝的EFA軟體搭配使用，並使用可EFA設定已啟用網路介面的啟動範本。

(選擇性) 測試 EFA

您可以執行包含在EFA軟體安裝中的fi_pingpong程式，來示範運算節點群組中兩個節點之間EFA啟用的通訊功能。如果此測試成功，則可能已EFA正確配置。

若要開始，您需要在運算節點群組中有兩個執行中執行個體。如果您的計算節點群組使用靜態容量，應該已經有可用的執行個體。對於使用動態容量的計算節點群組，您可以使用salloc指令啟動兩個節點。以下是叢集中的範例，其中動態節點群組名為hpc7g與名為的佇列相關聯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
```

使用找出兩個已配置節點的 IP 位址scontrol。在下面的範例中，地址是「10.3.140.69用於」hpc7g-1 和「10.3.132.211用於」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
```

使用 (或hpc7g-1) Connect 至其中一個節點 SSH (在此範例中為SSM)。請注意，這是一個內部 IP 地址，因此如果您使用，則可能需要從其中一個登錄節點進行連接SSH。另外請注意，執行個體必須透過運算節點群組啟動範本來設定SSH金鑰。

```
% ssh ec2-user@10.3.140.69
```

現在，fi_pingpong在服務器模式下啟動。

```
/opt/amazon/efa/bin/fi_pingpong -p efa
```

Connect 至第二個執行個體 (hpc7g-2)。

```
% ssh ec2-user@10.3.132.211
```

fi_pingpong在客戶端模式下運行，連接到服務器hpc7g-1。您應該會看到類似下列範例的輸出。

```
% /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
```

(選擇性) 使用 CloudFormation範本建立已啟用的EFA啟動範本

由於需要設定數個相依性EFA，因此已提供 CloudFormation 範本供您用來設定計算節點群組。它支援最多具有四張網路卡的執行個體。若要進一步了解具有多張網路卡的執行個體，請參閱 Amazon 彈性運算雲端使用者指南中的彈性網路界面。

從下列項目下載 CloudFormation 範本URL，然後將範本上傳至您使用的 CloudFormation AWS 區域主控台 AWS PCS。

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

在 AWS CloudFormation 主控台中開啟範本的情況下，輸入下列值。請注意，範本會提供一些預設參數值，您可以將它們保留為預設值。

- 在「提供堆疊名稱」下
  - 在「堆疊名稱」下，輸入描述性名稱。我們建議您合併為 AWS PCS計算節點群組選擇的名稱，例如*NODEGROUPNAME*-efa-lt。

- 在參數之下
  - 在下方 NumberOfNetworkCards，選擇節點群組中執行個體中的網路卡數量。
  - 在下 VpcId，選擇 AWS PCS叢集VPC的部署位置。
  - 在下 NodeGroupSubnetId，選擇叢集中要啟動EFA已啟VPC用執行個體的子網路。
  - 在下 PlacementGroupName，將欄位保留空白，為節點群組建立新的叢集置放群組。如果您有要使用的現有放置群組，請在此輸入其名稱。
  - 在下方 ClusterSecurityGroupId，選擇您要用來允許存取叢集中其他執行個體的安全性群組，以及 AWS PCSAPI. 許多客戶會從叢集中選擇預設安全性群組VPC。
  - 在下 SshSecurityGroupId，提供您用來允許對叢集中節點輸入SSH存取之安全性群組的 ID。
  - SSH對於 SshKeyName，選取要存取叢集中節點的金鑰配對。
  - 在中 LaunchTemplateName，輸入啟動範本的描述性名稱，例如*NODEGROUPNAME*`-efa-lt`。該名稱對於您要使用的 AWS 帳戶 位 AWS 區域 置而言必須是唯一的 AWS PCS。
- 在功能之下
  - 選中我確認 AWS CloudFormation 可能會創建IAM資源的複選框。

監視 CloudFormation 堆疊的狀態。當它到達`CREATE_COMPLETE`啟動模板就可以使用了。如上所述，將其與 AWS PCS計算節點群組搭配使用建立或更新計算節點群組。

使用網路檔案系統 AWS PCS

您可以將網路儲存磁碟區附加到「 AWS 平行運算服務」(AWS PCS) 計算節點群組中啟動的節點，以提供可寫入和存取資料和檔案的永久位置。您可以使用 AWS 服務提供的磁碟區。卷包括 Amazon Elastic File System（ AmazonEFS ），Amazon FSx NetApp ONTAP，Amazon FSx 打開 ZFS，Amazon FSx 的 Lustre 和 Amazon 文件緩存。您也可以使用自我管理的磁碟區，例如NFS伺服器。

本主題涵蓋搭配使用網路檔案系統的注意事項和範例 AWS PCS。

使用網路檔案系統的考量

各種檔案系統的實作細節不同，但也有一些常見的考量。

- 執行個體上必須安裝相關的檔案系統軟體。例如，要將 Amazon 用FSx於 Lustre，應該存在適當的Lustre軟件包。這可以通過將其包含在計算節點組中AMI或使用在實例啟動時運行的腳本來完成。
- 共用儲存磁碟區和計算節點群組執行個體之間必須有網路路由。

- 共用儲存磁碟區和計算節點群組執行個體上的安全群組規則必須允許連線至相關連接埠。
- 您必須在存取檔案系統的資源之間維持一致的POSIX使用者和群組命名空間。否則，在PCS叢集上執行的作業和互動式程序可能會遇到權限錯誤。
- 檔案系統掛載是使用EC2啟動範本完成的。掛載網路檔案系統時發生錯誤或逾時，可能會導致執行個體無法用於執行工作。反過來，這可能會導致意想不到的成本。如需偵錯啟動範本的詳細資訊，請參閱使用 Amazon EC2 啟動模板 AWS PCS。

網路掛載範例

您可以使用 AmazonEFS，Amazon FSx 為 Lustre，Amazon 打開ZFS和 Amazon FSx 文件緩存創建文件系統。展開下面的相關章節以查看每個網路掛載的範例。

Amazon EFS

檔案系統設定

創建一個 Amazon EFS 文件系統。確定它在每個可用區域中都有一個掛載目標，您將在此啟動PCS計算節點群組執行個體。同時確保每個掛載目標都與一個安全群組相關聯，該群組允許從PCS計算節點群組執行個體進行輸入和輸出存取。如需詳細資訊，請參閱 Amazon Elastic File System 使用者指南中的掛接目標和安全群組。

啟動範本

將檔案系統設定中的安全群組新增至您將用於計算節點群組的啟動範本。

包含使用`cloud-config`機制掛接 Amazon EFS 檔案系統的使用者資料。以您自己的詳細資料取代此指令碼中的下列值：

- *`mount-point-directory`*— 您將在其中安裝 Amazon 的每個實例上的路徑 EFS
- *`filesystem-id`*— 檔案系統的EFS檔案系統 ID

```
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 的光澤

檔案系統設定

在您將使FSx用的VPC位置創建一個 Lustre 文件系統。 AWS PCS若要盡量減少區域間傳輸，請部署在相同可用區域的子網路中，您將啟動大部分的PCS運算節點群組執行個體。確定檔案系統與允許PCS運算節點群組執行個體輸入和輸出存取的安全性群組相關聯。如需有關安全群組的詳細資訊，請參閱 Amazon 的 Lustre 使用者指南VPC中的使用 Amazon FSx 進行檔案系統存取控制。

啟動範本

包括用cloud-config來掛載 Lustre 檔案系統FSx的使用者資料。以您自己的詳細資料取代此指令碼中的下列值：

- *mount-point-directory*— 您要安裝 FSx Lustre 的執行個體上的路徑
- *filesystem-id*— Lustre 檔案系統的FSx檔案系統識別碼
- *mount-name*— Lustre 檔案系統FSx的掛載名稱
- *region-code*— Lustre 檔案系統的部署位置 (必須與您 AWS PCS的系統相同) AWS 區域 FSx
- (選擇性) *latest* — 任何Lustre支援 Lustre FSx 的版本

```
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 開放 ZFS

檔案系統設定

在您要使FSx用的VPC位置建立 For Open ZFS 檔案系統 AWS PCS。若要盡量減少區域間傳輸，請部署在相同可用區域的子網路中，您將啟動大部分的 AWS PCS運算節點群組執行個體。請確定檔案系統與允許 AWS PCS運算節點群組執行個體輸入和輸出存取的安全性群組相關聯。有關安全群組的詳細資訊，請參閱《開放式ZFS使用者指南》VPC中的《使FSx用 Amazon 管理檔案系統存取》。

啟動範本

包括用`cloud-config`於掛載 Open ZFS 檔案系統之根磁碟區FSx的使用者資料。以您自己的詳細資料取代此指令碼中的下列值：

- *`mount-point-directory`*— 您想要掛載 Open ZFS 共用的執行個體上FSx的路徑
- *`filesystem-id`*— 開啟檔案系統FSx的ZFS檔案系統 ID
- *`region-code`*— 用 AWS 區域 FSx於開啟ZFS檔案系統的部署位置 (必須與您的 AWS PCS系統相同)

```
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 文件緩存

檔案系統設定

在您要使用的VPC位置建立 Amazon 檔案快取 AWS PCS。若要盡量減少區域間傳輸，請在同一可用區域中選擇一個子網路，以啟動大部分PCS運算節點群組執行個體。確定檔案快取與安全性群組相關

聯，該群組允許PCS執行個體和檔案快取之間通訊埠 988 的輸入和輸出流量。如需有關安全群組的詳細資訊，請參閱 Amazon 檔案快取使用者指南VPC中的使用 Amazon 進行快取存取控制。

啟動範本

將檔案系統設定中的安全群組新增至您將用於計算節點群組的啟動範本。

包括用cloud-config來掛載 Amazon 檔案快取的使用者資料。以您自己的詳細資料取代此指令碼中的下列值：

- *mount-point-directory*— 您要安裝 FSx Lustre 的執行個體上的路徑
- *cache-dns-name*— 檔案快取的網域名稱系統 (DNS) 名稱
- *mount-name*— 檔案快取的掛載名稱

```
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 機器映像（AMIs）AWS PCS

AWS PCS與您提供AMIs的配置一起使用，在叢集中的節點上的軟體和配置提供極大的彈性。如果您正在嘗試 AWS PCS，則可以使用由AMI提供和維護的樣本 AWS。如果您在生產環境 AWS PCS中使用，我們建議您建立自己的產品AMIs。本主題涵蓋如何探索和使用範例AMIs，以及如何建置和使用您自己的自訂內容AMIs。

主題

- 使用示例 Amazon 機器映像（AMIs）AWS PCS
- 自定義 Amazon 機器映像（AMIs）AWS PCS
- 要建置自AMIs訂的軟體安裝程式 AWS PCS

使用示例 Amazon 機器映像（AMIs） AWS PCS

AWS提供範例 AMIs，您可以用來做為使用的起點 AWS PCS。

> **Important**
>
> 範例AMIs僅用於示範目的，不建議用於生產工作負載。

尋找目前的 AWS PCS樣本 AMIs

AWS Management Console

AWSPCS範例AMIs具有下列命名慣例：

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

接受的值

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

尋找 AWS PCS樣本 AMIs

1. 打開 Amazon EC2 控制台。
2. 導覽至AMIs。
3. 選擇公有映像。
4. 在AMI依屬性或標籤尋找中，AMI使用範本名稱搜尋。

   範例

   - 泥漿氣 23. AMI 11 支持重力子

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

   - x86 執行個體AMI的範例

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

> **Note**
>
> 如果有多個AMIs，請使AMI用最近的時間戳記。

5. 建立或更新計算節點群組時，請使用 AMI ID。

AWS CLI

您可以AMI使用以下命令找到最新的 AWS PCS示例。Replace (取代) *region-code* 與您使用的 AWS 區域 地方 AWS PCS，例如us-east-1。

- ```
  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
  ```
- 阿姆斯特

  ```
  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
  ```

建立或更新計算節點群組時，請使用 AMI ID。

進一步了解 AWS PCS樣本 AMIs

若要檢視範例目前和舊版 AWS PCS本的內容、組態詳細資訊AMIs，請參閱範例版 AWS PCS本說明 AMIs。
  ```

建立自己的AMIs兼容 AWS PCS

若要瞭解如何建立您自己AMIs的工作 AWS PCS，請參閱自定義 Amazon 機器映像（AMIs） AWS PCS。

自定義 Amazon 機器映像（AMIs） AWS PCS

AWS PCS旨在與 Amazon 機器映像（AMI），你帶來的服務工作。只要已正確安裝 AWS PCS代理程式和相容版本的 Slurm，它們就AMIs可以安裝任意軟體和組態。您必須使用 AWS提供的安裝程式，才能在自訂上安裝 AWS PCS軟體。AMI我們建議您使用 AWS-提供的安裝程序在您的自定義安裝 Slurm，AMI但如果你願意，你可以自己安裝 Slurm（不推薦）。

> **Note**
>
> 如果您想嘗試 AWS PCS而不構建自定義AMI，則可以使用提AMI供的示例 AWS。如需詳細資訊，請參閱使用示例 Amazon 機器映像（AMIs） AWS PCS。

本教學課程可協助您建立AMI可與PCS運算節點群組搭配使用，以支援您HPC和 AI/ML 工作負載的能力。

主題

- 步驟 1 — 啟動臨時執行個體
- 步驟 2 — 安裝 AWS PCS代理程式
- 步驟 3 — 安裝思隆
- 步驟 4 — (選擇性) 安裝其他驅動程式、程式庫和應用程式軟體
- 步驟 5 — 創建AMI兼容 AWS PCS
- 步驟 6 — 將自訂AMI與 AWS PCS計算節點群組搭配使用
- 步驟 7-終止臨時實例

步驟 1 — 啟動臨時執行個體

啟動可用於安裝和設定 AWS PCS軟體和 Slurm 排程器的臨時執行個體。您可以使用此執行個體來建立AMI相容的 AWS PCS。

啟動暫時執行個體

1. 打開 Amazon EC2 控制台。

2. 在導覽窗格中，選擇「執行個體」，然後選擇「啟動執行個體」以開啟新的啟動例項精靈。
3. (選擇性) 在 [名稱和標籤] 區段中，提供執行個體的名稱，例如`PCS-AMI-instance`。該名稱將指派作為執行個體的資源標籤 (`Name=PCS-AMI-instance`)。
4. 在「應用程式和作業系統映像」區段中，AMI為其中一個支援的作業系統選取一個。
5. 在 Instance type (執行個體類型) 區段中，選取支援的執行個體類型。
6. 在 Key pair (金鑰對) 區段中，選取要用於執行個體的金鑰對。
7. 在「網路設定」區段中：
   - 對於防火牆 (安全群組)，請選擇 [選取現有安全群組]，然後選取允許執行個體輸入SSH存取權的安全性群組。
8. 在 儲存 區段中，根據需求設定磁碟區。請務必設定足夠的空間來安裝您自己的應用程式和程式庫。
9. 在 Summary (摘要) 面板中，選擇 Launch instance (啟動執行個體)。

步驟 2 — 安裝 AWS PCS代理程式

安裝代理程式，該代理程式會設定由 AWS PCS啟動的執行個體以搭配 Slurm 使用。

安裝 AWS PCS 代理

1. 連接至您啟動的執行個體。如需詳細資訊，請參閱 Connect 到 Linux 執行個體。
2. (選擇性) 為確保所有軟體套件皆為最新版本，請對執行個體執行快速軟體更新。此程序可能需要幾分鐘的時間。
   - Amazon Linux 2, RHEL 9, 洛基 Linux 9

     ```
     sudo yum update -y
     ```

   - Ubuntu

     ```
     sudo apt-get update && sudo apt-get upgrade -y
     ```

3. 重新啟動執行個體並重新連線至其中。
4. 下載 AWS PCS代理程式安裝檔案。安裝檔案會封裝成一個壓縮的 tarball (`.tar.gz`) 檔案。若要下載最新穩定版本，請使用下列命令：替代 *`region`* 與 AWS 區域 您啟動臨時實例的位置，例如`us-east-1`。

```
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
```

您也可以使用上述指令`latest`中的版本號碼取代版本號碼，以取得最新版本 (例如:`aws-pcs-agent-v1-latest.tar.gz`)。

> **Note**
>
> 這可能會在 AWS PCS代理程式軟體的 future 版本中變更。

5. （可選）驗證 AWS PCS軟件 tarball 的真實性和完整性。我們建議您執行這項操作來確認軟體發行者的身分，並檢查檔案自發行以來並未遭到變更或損毀。

   a. 下載的公開GPG金鑰 AWS PCS並將其匯入您的金鑰圈。替代 *region* 與您啟動臨時實例的 AWS 區域 位置。命令應傳回金鑰值。記錄索引鍵值；您可以在下一個步驟中使用它。

   ```
   wget https://aws-pcs-repo-public-keys-region.s3.amazonaws.com/aws-pcs-public-key.pub && \
       gpg --import aws-pcs-public-key.pub
   ```

   b. 執行下列命令以驗證GPG金鑰的指紋。

   ```
   gpg --fingerprint 7EEF030EDDF5C21C
   ```

   該命令應返回與以下內容相同的指紋：

   ```
   1C24 32C1 862F 64D1 F90A  239A 7EEF 030E DDF5 C21C
   ```

   > **Important**
   >
   > 如果指紋不相符，請勿執行 AWS PCS代理程式安裝指令碼。聯絡 AWS 支援。

   c. 下載簽名文件並驗證 AWS PCS軟件 tarball 文件的簽名。Replace (取代) *region* 與 AWS 區域 您啟動臨時實例的位置，例如`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
```

輸出格式應類似以下內容：

```
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
```

如果結果包含，Good signature且指紋與上一步中傳回的指紋相符，請繼續執行下一個步驟。

> **Important**
>
> 如果指紋不相符，請勿執行 AWS PCS軟體安裝指令碼。聯絡 AWS 支援。

6. 從壓縮檔案解壓縮.tar.gz檔案並導覽至解壓縮的目錄。

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

7. 安裝 AWS PCS 軟體。

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

8. 檢查 AWS PCS軟體版本檔案以確認安裝成功。

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

輸出格式應類似以下內容：

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

步驟 3 — 安裝思隆

安裝與之兼容的 Slurm 版本。 AWS PCS

要安裝思龍

1. Connect 至安裝軟體的相同暫存執行個 AWS PCS體。
2. 下載 Slurm 安裝程式軟體。Slurm 安裝程式會封裝成壓縮壓縮檔案 () `.tar.gz` 檔案。若要下載最新穩定版本，請使用下列命令：替代 *region* 使用您 AWS 區域 的臨時實例，例如`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
   ```

   您也可以使用上述指令`latest`中的版本號碼取代版本號碼，以取得最新版本 (例如:`aws-pcs-slurm-23.11-installer-latest.tar.gz`)。

   > **Note**
   >
   > 這可能會在 Slurm 安裝程式軟體的 future 版本中發生變更。

3. （可選）驗證 Slurm 安裝程序壓縮包的真實性和完整性。我們建議您執行這項操作來確認軟體發行者的身分，並檢查檔案自發行以來並未遭到變更或損毀。

   a. 下載的公開GPG金鑰 AWS PCS並將其匯入您的金鑰圈。替代 *region* 與您啟動臨時實例的 AWS 區域 位置。命令應傳回金鑰值。記錄索引鍵值；您可以在下一個步驟中使用它。

      ```
      wget https://aws-pcs-repo-public-keys-region.s3.amazonaws.com/aws-pcs-public-
      key.pub && \
          gpg --import aws-pcs-public-key.pub
      ```

   b. 執行下列命令以驗證GPG金鑰的指紋。

      ```
      gpg --fingerprint 7EEF030EDDF5C21C
      ```

      該命令應返回與以下內容相同的指紋：

      ```
      1C24 32C1 862F 64D1 F90A  239A 7EEF 030E DDF5 C21C
      ```

> **Important**
>
> 如果指紋不匹配，請不要運行 Slurm 安裝腳本。聯絡 AWS 支援。

c. 下載簽名文件並驗證 Slurm 安裝程序壓縮包文件的簽名。Replace (取代) *region* 與 AWS 區域 您啟動臨時實例的位置，例如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
```

輸出格式應類似以下內容：

```
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
```

如果結果包含，Good signature且指紋與上一步中傳回的指紋相符，請繼續執行下一個步驟。

> **Important**
>
> 如果指紋不匹配，請不要運行 Slurm 安裝腳本。聯絡 AWS 支援。

4. 從壓縮的 .tar.gz 檔案中解壓縮檔案，然後導覽至解壓縮的目錄。

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

5. 安裝思龍。安裝程式會下載、編譯和安裝 Slurm 及其相依性。這需要幾分鐘的時間，具體取決於您選取的臨時執行個體的規格。

```
sudo ./installer.sh -y
```

6. 檢查排程器版本檔案以確認安裝。

```
cat /opt/aws/pcs/scheduler/slurm-23.11/version
```

輸出格式應類似以下內容：

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

步驟 4 — (選擇性) 安裝其他驅動程式、程式庫和應用程式軟體

在暫存執行個體上安裝其他驅動程式、程式庫和應用程式軟體。安裝程序會根據特定的應用程式和程式庫而有所不同。如果您 AWS PCS之前尚未建立自AMI訂項目，我們建議您先建置並測試只安裝 AWS PCS軟體和 Slurm，然後在確認初始成功後逐步新增您自己的軟體和組態。AMI

範例

- 彈性織物適配器（EFA）軟件。如需詳細資訊，請參閱 Amazon 彈性運算雲端使用者指南EC2中的 Amazon 上的HPC工作負載入門EFA和MPI使用工作負載。
- Amazon Elastic File System（AmazonEFS）客戶端。如需詳細資訊，請參閱 Amazon 彈性檔案系統使用者指南中的手動安裝 Amazon 用EFS戶端。
- 光澤的客戶，使用 Amazon 的 Lustre 和 Amazon FSx 文件緩存。如需詳細資訊，請參閱《Lustre 使用者指南》中的〈安裝 Lustr e 用戶端〉。FSx
- Amazon CloudWatch 代理，使用 CloudWatch 日誌和指標。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的安裝 CloudWatch 代理程式。
- AWS 神經元，使用 trn* 和 inf * 執行個體類型。有關更多信息，請參閱AWS 神經元文檔。
- NVIDIA驅動程式CUDA、、和DCGM，可使用 p* 或 g* 執行個體類型。

步驟 5 — 創建AMI兼容 AWS PCS

安裝必要的軟體元件之後，您可以建立可重複使用AMI的軟體元件，以便在 AWS PCS計算節點群組中啟動執行個體。

若要AMI從暫存執行個體建立

1. 打開 Amazon EC2 控制台。

2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體。選擇動作、影像、建立映像。
4. 對於 Create image (建立映像)，執行下列動作：
   a. 在「影像名稱」中，輸入的描述性名稱AMI。
   b. (選擇性) 在影像說明中，輸入用途的簡短描述AMI。
   c. 選擇 Create image (建立映像)。
5. 在導覽窗格中，選擇AMIs。
6. 找到您在列表中創建的 AMI tht。等待其狀態從 [擱置中] 變更為 [可用]，然後搭配 AWS PCS運算節點群組使用。

步驟 6 — 將自訂AMI與 AWS PCS計算節點群組搭配使用

您可以將自訂AMI與新的或現有的 AWS PCS計算節點群組搭配使用。

New compute node group

若要使用自訂 AMI

1. 開啟主AWS PCS控台。
2. 在導覽窗格中，選擇叢集。
3. 選擇您要使用自訂的叢集AMI，然後選取計算節點群組。
4. 建立新的計算節點群組。如需詳細資訊，請參閱在中建立計算節點群組 AWS PCS。在 AMIID 下，搜尋您要使用之自訂AMI的名稱或 ID。完成計算節點群組的設定，然後選擇 [建立計算節點群組]。
5. (選擇性) 確認AMI支援執行個體啟動。啟動運算節點群組中的執行個體。您可以將計算節點群組設定為具有單一靜態執行個體，或將工作提交至使用計算節點群組的佇列。
   a. 檢查 Amazon 主EC2控台，直到執行個體出現標有新運算節點群組 ID 的標記為止。有關此方面的更多信息，請參閱尋找運算節點群組執行個體 AWS PCS..
   b. 當您看到執行個體啟動並完成其啟動程序時，請確認其使用的是預期的AMI。若要這麼做，請選取執行個體，然後在 [詳細資料] 下檢查 AMIID。它應該符合AMI您在計算節點群組設定中設定的。
   c. (選擇性) 將計算節點群組擴展配置更新為您偏好的值。

Existing compute node group

若要使用自訂 AMI

1. 開啟主AWS PCS控台。
2. 在導覽窗格中，選擇叢集。
3. 選擇您要使用自訂的叢集AMI，然後選取計算節點群組。
4. 選取您要設定的節點群組，然後選擇 [編輯]。在 AMIID 下，搜尋您要使用之自訂AMI的名稱或 ID。完成計算節點群組的設定，然後選擇 [更新]。在計算節點群組中啟動的新執行個體將使用更新的 AMI ID。現有實例將繼續使用舊實例，AMI直到 AWS PCS替換它們為止。如需詳細資訊，請參閱更新計 AWS PCS算節點群組。
5. (選擇性) 確認AMI支援執行個體啟動。啟動運算節點群組中的執行個體。您可以將計算節點群組設定為具有單一靜態執行個體，或將工作提交至使用計算節點群組的佇列。

   a. 檢查 Amazon 主EC2控台，直到執行個體出現標有新運算節點群組 ID 的標記為止。有關此方面的更多信息，請參閱尋找運算節點群組執行個體 AWS PCS..

   b. 當您看到執行個體啟動並完成其啟動程序時，請確認其使用的是預期的AMI。若要這麼做，請選取執行個體，然後在 [詳細資料] 下檢查 AMIID。它應該符合AMI您在計算節點群組設定中設定的。

   c. (選擇性) 將計算節點群組擴展配置更新為您偏好的值。

步驟 7-終止臨時實例

確認您的AMI作品如預期 AWS PCS，您可以終止暫時執行個體，以停止向其產生費用。

終止暫時執行個體

1. 打開 Amazon EC2 控制台。
2. 在導覽窗格中，選擇 Instances (執行個體)。
3. 選取您建立的暫時執行個體，然後選擇「動作」、「執行處理」狀態、「終止」
4. 系統提示您確認時，請選擇「終止」。

要建置自AMIs訂的軟體安裝程式 AWS PCS

AWS 提供可下載的檔案，可在執行個 AWS PCS體上安裝軟體。 AWS 還提供了可以下載，編譯和安裝 Slurm 及其依賴關係的相關版本的軟件。您可以使用這些指示來建立自訂以AMIs供搭配使用， AWS PCS也可以使用自己的方法。

內容

AWS PCS軟件安裝

AWS PCS軟體安裝程式會設定執行個體，以便在執行個體啟動程序 AWS PCS期間使用。您必須使用AWS提供的安裝程式，才能在自訂上安裝 AWS PCS軟體。AMI

思盧姆安裝程序

Slurm 安裝程序下載，編譯和安裝 Slurm 及其依賴項的相關版本。您可以使用 Slurm 安裝程式AMIs為. AWS PCS 如果您自己的機制與 Slurm 安裝程式提供的軟體配置一致，您也可以使用它們。

AWS提供的軟體會安裝下列項目：

- Slurm 在要求的主要和維護版本（目前的版本 23.11.8）-許可證 2 GPL
  - 思盧姆是建立與設`--sysconfdir`置為 `/etc/slurm`
  - 思盧姆是建立與選項和 `--enable-pam --without-munge`
  - 思盧姆是建立與選項 `--sharedstatedir=/run/slurm/`
  - 思盧姆是建立與PMIX支持 JWT
  - 思倫安裝在 `/opt/aws/pcs/schedulers/slurm-23.11`
- 開放 PMIX（版本 4.2.6）-許可證
  - 打開PMIX被安裝為子目錄 `/opt/aws/pcs/scheduler/`
- 西班牙語庫（版本 1.15.3）-許可證 -2.0 MPL

- libjwt 被安裝為子目錄 `/opt/aws/pcs/scheduler/`

AWS提供的軟體會變更系統組態，如下所示：

- 由組建建立的 Slurm `systemd` 檔案會以檔案名稱複`/etc/systemd/system/`製到。`slurmd-23.11.service`
- 如果它們不存在，則使用UID/GID的創建 Slurm 用戶和組（`slurm:slurm`）。`401`
- 在 Amazon Linux 2 和岩石 Linux 9 上，安裝程序添加了EPEL存儲庫以安裝所需的軟件以構建 Slurm 或其依賴項。
- RHEL9在安裝將啟用`codeready-builder-for-rhel-9-rhui-rpms`並`epel-release-latest-9`從`fedoraproject`安裝所需的軟件來構建 Slurm 或其依賴關係。

支援的作業系統

該 AWS PCS軟件和 Slurm 安裝程序支持以下操作系統：

- Amazon Linux 2
- RedHat 企業版 9
- 洛基 Linux 9
- Ubuntu

> **Note**
>
> AWS 深度學習 AMIs（DLAMI）基於 Amazon Linux 2 和 Ubuntu 22.04 的版本應該與 AWS PCS軟件和思樂安裝程序兼容。如需詳細資訊，請參閱AWS 深度學習 AMIs 開發人員指南 DLAMI中的「選擇您的」。

支援的執行個體類型

AWS PCS軟體和 Slurm 安裝程式支援任何 x86_64 或 arm64 執行個體類型，而不是可以執行其中一個支援的作業系統。

支持的思倫版本

支持以下主要版本的思倫：

- 泥漿

使用總和檢查碼驗證安裝程式

您可以使用SHA256總和檢查碼來驗證安裝程式 tarball (.tar.gz) 檔案。我們建議您執行這項操作來確認軟體發行者的身分識別，並檢查應用程式自發行以來並未遭到變更或損毀。

驗證壓縮包的步驟

使用 sha256sum 公用程式做為總和SHA256檢查碼，並指定壓縮標籤檔案名稱。您必須從儲存 tarball 檔案的目錄執行指令。

- SHA256

```
$ sha256sum tarball_filename.tar.gz
```

命令應以下列格式傳回總和檢查碼值。

```
checksum_value tarball_filename.tar.gz
```

將命令傳回的總和檢查碼值與下表中提供的總和檢查碼值做比較。如果總和檢查碼相符，則執行安裝指令碼是安全的。

> **Important**
>
> 如果總和檢查碼不相符，請勿執行安裝指令碼。請聯絡 AWS Support。

例如，下列指令會產生 Slurm 23.11.9 壓縮包的SHA256總和檢查碼。

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

輸出範例：

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

下表列出安裝程式最新版本的總和檢查碼。Replace (取代) *us-east-1* 與你使 AWS 區域 用的地方 AWS PCS。

Installer (安裝程式)	下載 URL	SHA256校驗和
思隆	`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 PCS代理商	`https://aws-pcs-repo- us-east-1 .s3.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.0.0-1.tar.gz`	`d2d3d68d00c685435c38af471d7e2492dde5ce9eb222d7b6ef0042144b134ce0`

思盧姆版本 AWS PCS

SchedMD 透過新功能、最佳化和安全性修補程式持續增強 Slurm。SchedMD 會定期發行新的主要版本，並計劃在任何給定時間支援最多 3 個版本。 AWS PCS最初支持思盧姆 23.11。您可以在新版本發布後升級 Slurm 主要版本。 AWS PCS旨在使用補丁版本自動更新 Slurm 控制器。

當 SchedMD 結束對特定主要版本的支援時， AWS PCS也會終止對該主要版本的支援。 AWS PCS如果 Slurm 主要版本接近其生命週期結束，則會傳送預先通知，以協助客戶知道何時將叢集升級至較新的受支援版本。

我們建議您使用支援的最新 Slurm 版本來部署叢集，以存取最新的進步和改進。

關於思盧姆版本的常見問題

Slurm 版本 AWS PCS支持多長時間？

AWS PCS遵循主要版本的 ScheDMD 支援週期。 AWS PCS在任何給定時間支持多達 3 個主要版本。ScheDMD 發行新的主要版本之後， AWS PCS淘汰最舊的支援版本。 AWS PCS儘快發佈 Slurm 的新主要版本，但 SchedMD 發行版本與其在. AWS PCS

什麼時候 AWS PCS通知我 Slurm 版本的 Support 生命終止（EOSL）？

AWS PCS在日期之前，以預先確定的節奏多次通知您。EOSL

當 Slurm 版本接近時，我該怎麼辦？EOSL

您必須先更新 Slurm 版本，EOSL以協助維護安全且受支援的環境。

如何更新我的叢集以使用新的主要版本的 Slurm？

若要更新 Slurm 版本，您必須建立新叢集。您也必須升級至中的對等 AWS PCS軟體，AMI並使用它來建立新叢集的運算節點群組。

我的叢集將如何取得新的 Slurm 修補程式版本？

AWS PCS旨在自動套用修補程式以解決 Slurm 常見弱點和曝光 ()。CVEs AWS PCS將修補程式套用至在內部服務擁有的帳戶中執行的叢集控制器。您必須使用 AWS Management Console 或動 AWS PCSAPI作在您的EC2執行個體上安裝修補程式 AWS 帳戶。

如果我沒有在EOSL日期之前更新 Slurm 怎麼辦？

AWS PCS旨在停止具有不支援 Slurm 版本的叢集。您必須更新叢集控制器的 Slurm 主要版本，以及計算節點群組上安裝的 AWS PCS軟體。

支援多少 Slurm 版本 AWS PCS？

AWS PCS在任何給定時間支持多達 3 個主要 Slurm 版本，包括當前版本和 2 個以前的主要版本。

我應該申請哪些 Slurm 版本更新？

我們強烈建議您在叢集中的所有元件上使用相同的主要版本，並在發行最新的修補程式後立即安裝。您AMIs的計算節點群組必須使用與 Slurm 版本的叢集控制器相容的 Slurm 軟體版本。您的 Slurm 主要版本AMIs必須在群集控制器上的 Slurm 主要版本的 2 個版本之內。安裝在叢集中AMI和執行中EC2執行個體上的 Slurm 版本不能比叢集控制器上的 Slurm 版本新。若要維護叢集的支援，您AMIs必須使用支援的 AWS PCS軟體版本。

如果我更新 Slurm 主要版本，但在計算節點群組中使用較舊的 Slurm 軟體，該怎AMI麼辦？

您必須將 AWS PCS軟體更新至相同版本才能使用新的 Slurm 功能。為了獲得完全 AWS PCS支持，所有 Slurm 組件都必須使用支持的版本。綜上所述：

- 當您的群集控制器和所有組件（AWS PCS包） AWS 帳戶 都使用受支持的版本時，我們能夠提供全面支持。
- AWS PCS被設計為在其控制器的 Slurm 版本到達時停止集群。EOSL
- 如果 AWS 帳戶 觸及範圍內的 Slurm 版本元件EOSL，則不支援您的叢集。

我應該以何種順序更新叢集中的元件？

您必須先更新叢集控制器的 Slurm 版本，才能使用較新的 Slurm 版本。AMI您可以更新計算節點群組以使用AMI. AWS PCS使用啟AMI動計算節點群組中的新EC2執行個體。 AWS PCS不會更新具有EC2執行中工作的現有執行個體；其設計目的 AWS PCS是在其工作完成後終止這些執行個體。

是否 AWS PCS為 Slurm 版本提供擴展支持？

沒有 我們會傳達有關延伸支援選項的詳細資訊，包括任何額外費用和所提供的特定支援範圍。

AWS 平行運算服務的安全性

雲安全 AWS 是最高的優先級。身為 AWS 客戶，您可以從資料中心和網路架構中獲益，這些架構是為了滿足對安全性最敏感的組織的需求而建置的。

安全是 AWS 與您之間共同承擔的責任。共同責任模型將其描述為雲端的安全性和雲端中的安全性：

- 雲端安全性 — AWS 負責保護中執行 AWS 服務的基礎架構 AWS 雲端。 AWS 還為您提供可以安全使用的服務。若要瞭解適用於「 AWS 平行運算服務」的法規遵循計劃，請參閱合規計劃AWS 服務範圍)。
- 雲端中的安全性 — 您的責任取決於您使用的 AWS 服務。您也必須對其他因素負責，包括資料的機密性、您公司的要求和適用法律和法規。

本文件可協助您瞭解如何在使用時套用共同責任模型 AWS PCS。下列主題說明如何設定 AWS PCS 以符合安全性與合規性目標。您也會學到如何使用其他可協助您監控和保護 AWS PCS資源的 AWS 服務。

主題

- AWS 平行運算服務中的資料保護
- 使用介面端點存取 AWS 平行運算服務 (AWS PrivateLink)
- AWS 平行運算服務的 Identity and Access Management
- AWS 平行運算服務的合規性驗證
- AWS 平行運算服務的彈性
- AWS 平行運算服務的基礎架構安全
- AWS 並行計算服務中的漏洞分析和管理
- 預防跨服務混淆代理人
- AWS 平行運算服務的安全性最佳作法

AWS 平行運算服務中的資料保護

AWS 共同責任模式適用於 AWS 平行運算服務中的資料保護。如此模型中所述， AWS 負責保護執行所有 AWS 雲端. 您負責維護在此基礎設施上託管內容的控制權。您也同時負責所使用 AWS 服務 的安全組態和管理任務。如需有關資料隱私權的詳細資訊，請參閱資料隱私權FAQ。 如需歐洲資料保護的相關資訊，請參閱AWS 安全性GDPR部落格上的AWS 共同責任模型和部落格文章。

基於資料保護目的，我們建議您使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 保護 AWS 帳戶 認證並設定個別使用者。如此一來，每個使用者都只會獲得授與完成其任務所必須的許可。我們也建議您採用下列方式保護資料：

- 對每個帳戶使用多重要素驗證 (MFA)。
- 使用SSL/TLS與 AWS 資源溝通。我們需要 TLS 1.2 並推薦 TLS 1.3。
- 使用設定API和使用者活動記錄 AWS CloudTrail。
- 使用 AWS 加密解決方案以及其中的所有默認安全控制 AWS 服務。
- 使用進階的受管安全服務 (例如 Amazon Macie)，協助探索和保護儲存在 Amazon S3 的敏感資料。
- 如果 AWS 透過命令列介面或存取時需要 FIPS 140-3 驗證的密碼編譯模組API，請使用端點。FIPS 如需有關可用FIPS端點的詳細資訊，請參閱聯邦資訊處理標準 (FIPS) 140-3。

我們強烈建議您絕對不要將客戶的電子郵件地址等機密或敏感資訊，放在標籤或自由格式的文字欄位中，例如名稱欄位。這包括當您與控制台、API、 AWS PCS或一起 AWS 服務 使用或使用其他控制台時 AWS SDKs。 AWS CLI您在標籤或自由格式文字欄位中輸入的任何資料都可能用於計費或診斷日誌。如果您提供URL給外部伺服器，我們強烈建議您不要在中包含認證資訊，URL以驗證您對該伺服器的要求。

靜態加密

當您使用 AWS Management Console、 AWS CLI或建立 AWS 平行運算服務 (AWS PCS) 叢集時，預設會為靜態資料啟用加密 AWS SDKs。 AWS PCS API AWS PCS使用AWS 擁有的密KMS鑰來加密靜態數據。如需詳細資訊，請參閱AWS KMS 開發人員指南中的客戶 AWS 金鑰和金鑰。叢集密碼會儲存在中， AWS Secrets Manager 並使用秘密管理員管理的KMS金鑰加密。如需詳細資訊，請參閱使用中的叢集密碼 AWS PCS。

在 AWS PCS叢集中，下列資料處於靜態狀態：

- 排程器狀態 — 包括叢集中執行作業和佈建節點的資料。這是 Slurm 持續存在於您定`StateSaveLocation`義的. `slurm.conf` 如需詳細資訊，請參閱 Slurm 文件StateSaveLocation中的說明。 AWS PCS在工作完成後刪除工作資料。
- 排程器驗證密碼 — AWS PCS 使用它來驗證叢集中的所有排程器通訊。

對於排程器狀態資訊，在將資料和中繼資料寫入檔案系統之前， AWS PCS會自動加密資料和中繼資料。加密的檔案系統使用業界標準 AES -256 加密演算法處理靜態資料。

傳輸中加密

無論您 AWS PCSAPI使用 AWS Command Line Interface (AWS CLI) 還是使用簽名版本 4 簽名過程的TLS加密連接 AWS SDKs。如需詳細資訊，請參閱AWS Identity and Access Management 使用者指南中的簽署 AWS API要求。 AWS 透過使用您用來連線之安全認證的IAM原則來管理存取控制。API

AWS PCS用TLS於連接到其他 AWS 服務。

在 Slurm 叢集中，排程器會使用`auth/slurm`驗證外掛程式設定，該外掛程式可為所有排程器通訊提供驗證。Slurm 不會在應用程式層級為其通訊提供加密，所有跨叢集執行個體流動的資料都會保留在本機，因此如果這些執行個體支援傳輸中的VPC加密，則會受到加密。EC2 VPC如需詳細資訊，請參閱 Amazon 彈性運算雲端使用者指南中的傳輸中加密。帳戶中叢集節點之控制器 (在服務帳戶中佈建) 之間的通訊會加密。

金鑰管理

AWS PCS使用AWS 擁有的KMS金鑰來加密資料。如需詳細資訊，請參閱AWS KMS 開發人員指南中的客戶 AWS 金鑰和金鑰。叢集密碼會儲存在中， AWS Secrets Manager 並使用秘密管理員管理的KMS金鑰加密。如需詳細資訊，請參閱使用中的叢集密碼 AWS PCS。

網際網路流量隱私權

AWS PCS叢集的計算資源位於客戶帳戶VPC中的 1 內。因此，叢集內的所有內部 AWS PCS服務流量都會保留在 AWS 網路內，而且不會透過網際網路傳輸。用戶和 AWS PCS節點之間的通信可以通過互聯網傳輸，我們建議使用SSH或 Systems Manager 連接到節點。如需詳細資訊，請參閱什麼是 AWS Systems Manager？ 在《AWS Systems Manager 使用者指南》中。

您也可以使用下列供應項目將內部部署網路連線到 AWS：

- AWS Site-to-Site VPN。 如需詳細資訊，請參閱什麼是 AWS Site-to-Site VPN？ 在《AWS Site-to-Site VPN 使用者指南》中。
- 一個 AWS Direct Connect. 如需詳細資訊，請參閱什麼是 AWS Direct Connect？ 在《AWS Direct Connect 使用者指南》中。

您可以存取 AWS PCSAPI以執行服務的管理工作。您和您的使用者存取 Slurm 端點連接埠，以直接與排程器互動。

加密流量 API

若要存取 AWS PCSAPI，用戶端必須支援傳輸層安全性 (TLS) 1.2 或更新版本。我們需要 TLS 1.2 並推薦 TLS 1.3。客戶還必須支持具有完美前向保密（ ）的密碼套件，例如短暫的迪菲-赫爾曼（PFS）或橢圓曲線迪菲-赫爾曼短暫（ ）。DHE ECDHE現代系統(如 Java 7 和更新版本)大多會支援這些模式。此外，請求必須使用存取金鑰 ID 和與IAM主體相關聯的秘密存取金鑰來簽署。您也可以使用 AWS Security Token Service (AWS STS) 來產生用來簽署要求的臨時安全登入資料。

加密資料流量

從存取排程器端點的支援EC2執行個體，以及從中存取執行個體之間的 ComputeNodeGroup 執行個體啟用傳輸中資料加密 AWS 雲端。如需詳細資訊，請參閱傳輸中加密。

使用介面端點存取 AWS 平行運算服務 (AWS PrivateLink)

您可以使 AWS PrivateLink 用在您VPC與 AWS 平行運算服務 (AWS PCS) 之間建立私人連線。您可以 AWS PCS像在您的一樣訪問VPC，而無需使用 Internet 網關，NAT設備，VPN連接或 AWS Direct Connect 連接。您中的執行個體VPC不需要公用 IP 位址即可存取 AWS PCS。

您可以建立由 AWS PrivateLink提供支援的介面端點來建立此私有連線。我們會在您為介面端點啟用的每個子網中建立端點網路介面。這些是由要求者管理的網路介面，可做為目的地流量的進入點。 AWS PCS

如需詳細資訊，請參閱AWS PrivateLink 指南 AWS PrivateLink中的 AWS 服務 透過存取。

的注意事項 AWS PCS

設定的介面端點之前 AWS PCS，請先檢閱AWS PrivateLink 指南中的使用介面VPC端點存取AWS服務。

AWS PCS支援透過介面端點呼叫其所有API動作。

如果您VPC沒有直接存取網際網路，則必須設定VPC端點，以啟用計算節點群組執行個體呼叫 AWS PCS`RegisterComputeNodeGroupInstance`API動作。

建立的介面端點 AWS PCS

您可以建立 AWS PCS使用 Amazon VPC 主控台或 AWS Command Line Interface (AWS CLI) 的介面端點。如需詳細資訊，請參閱《AWS PrivateLink 指南》中的建立介面端點。

建立 AWS PCS使用下列服務名稱的介面端點：

```
com.amazonaws.region.pcs
```

Replace (取代) *region* 具有要在中建 AWS 區域 立端點的識別碼，例如us-east-1。

如果您DNS為介面端點啟用 private，您可以 AWS PCS使用其預設地區DNS名稱提出API要求。例如：pcs.us-east-1.amazonaws.com。

為您的介面端點建立端點政策

端點策略是您可以附加到介面端點的IAM資源。預設端點策略允許 AWS PCS透過介面端點進行完整存取。若要控制 AWS PCS從您的允許存取VPC，請將自訂端點原則附加到介面端點。

端點政策會指定以下資訊：

- 可以執行動作 (AWS 帳戶、IAM使用者和IAM角色) 的主參與者。
- 可執行的動作。
- 可供執行動作的資源。

如需詳細資訊，請參閱《AWS PrivateLink 指南》中的使用端點政策控制對服務的存取。

範例： AWS PCS處理行動的VPC端點策略

以下是自訂端點政策的範例。當您將此原則附加至介面端點時，它會授與具有指定之叢集之所有主體所列 AWS PCS動作的存取權 *cluster-id*。 取代 *region* 具有叢集 AWS 區域 的識別碼，例如us-east-1。Replace (取代) *account-id* 與群集的 AWS 帳戶 數量。

```
{
    "Statement": [
            {
                "Action": [
                "pcs:CreateCluster",
                "pcs:ListClusters",
                "pcs:DeleteCluster",
                "pcs:GetCluster",
                ],
                "Effect": "Allow",
                "Principal": "*",
                "Resource": [
                    "arn:aws:pcs:region:account-id:cluster/cluster-id*"
                ]
            }
```

```
        ]
}
```

AWS 平行運算服務的 Identity and Access Management

AWS Identity and Access Management (IAM) 可協助系統管理員安全地控制 AWS 資源存取權。 AWS 服務 IAM管理員控制誰可以驗證（登錄）和授權（有權限）使用 AWS PCS資源。IAM是一種您 AWS 服務 可以使用，無需額外費用。

主題

- 物件
- 使用身分驗證
- 使用政策管理存取權
- AWS 平行運算服務如何搭配使用 IAM
- AWS 平行運算服務的身分識別原則範例
- AWSAWS 平行運算服務的管理原則
- AWS PCS 的服務連結角色
- Amazon EC2 現貨角色 AWS PCS
- 的最低權限 AWS PCS
- IAM AWS 平行運算服務執行個體設定檔
- 疑難排解 AWS 平行運算服務識別與存取

物件

你如何使用 AWS Identity and Access Management （IAM）不同，具體取決於你在做的工作 AWS PCS。

服務使用者 — 如果您使用 AWS PCS服務執行工作，則管理員會為您提供所需的認證和權限。當您使用更多 AWS PCS功能來完成工作時，您可能需要其他權限。了解存取許可的管理方式可協助您向管理員請求正確的許可。如果無法存取中的圖徵 AWS PCS，請參閱疑難排解 AWS 平行運算服務識別與存取。

服務管理員 — 如果您負責公司的 AWS PCS資源，您可能擁有完整的存取權 AWS PCS。決定您的服務使用者應該存取哪些 AWS PCS功能和資源是您的工作。然後，您必須向IAM管理員提交請求，才能

變更服務使用者的權限。檢閱此頁面上的資訊，以瞭解的基本概念IAM。若要深入瞭解貴公司如何IAM搭配使用 AWS PCS，請參閱AWS 平行運算服務如何搭配使用 IAM。

IAM系統管理員 — 如果您是IAM系統管理員，您可能想要瞭解如何撰寫原則來管理存取權的詳細資訊AWS PCS。若要檢視可在中使用 AWS PCS的識別型原則範例IAM，請參閱。AWS 平行運算服務的身分識別原則範例

使用身分驗證

驗證是您 AWS 使用身分認證登入的方式。您必須以IAM使用者身分或假設IAM角色來驗證 (登入 AWS)。 AWS 帳戶根使用者

您可以使用透過 AWS 身分識別來源提供的認證，以聯合身分識別身分登入。 AWS IAM Identity Center (IAM身分識別中心) 使用者、貴公司的單一登入驗證，以及您的 Google 或 Facebook 認證都是聯合身分識別的範例。當您以同盟身分登入時，您的管理員先前會使用IAM角色設定聯合身分識別。當您使 AWS 用同盟存取時，您會間接擔任角色。

根據您的使用者類型，您可以登入 AWS Management Console 或 AWS 存取入口網站。如需登入的詳細資訊 AWS，請參閱AWS 登入 使用者指南中的如何登入您 AWS 帳戶的。

如果您 AWS 以程式設計方式存取，請 AWS 提供軟體開發套件 (SDK) 和命令列介面 (CLI)，以使用您的認證以密碼編譯方式簽署您的要求。如果您不使用 AWS 工具，則必須自行簽署要求。如需使用建議的方法自行簽署要求的詳細資訊，請參閱使用IAM者指南中的簽署 AWS API要求。

無論您使用何種身分驗證方法，您可能都需要提供額外的安全性資訊。例如， AWS 建議您使用多重要素驗證 (MFA) 來增加帳戶的安全性。若要深入瞭解，請參閱使用AWS IAM Identity Center 者指南中的多重要素驗證和使用多重要素驗證 (MFA) AWS的使用IAM者指南。

AWS 帳戶 根使用者

當您建立時 AWS 帳戶，您會從一個登入身分開始，該身分可完整存取該帳戶中的所有資源 AWS 服務和資源。此身分稱為 AWS 帳戶 root 使用者，可透過使用您用來建立帳戶的電子郵件地址和密碼登入來存取。強烈建議您不要以根使用者處理日常任務。保護您的根使用者憑證，並將其用來執行只能由根使用者執行的任務。如需需要您以 root 使用者身分登入的完整工作清單，請參閱《使用指南》中的〈需要 root 使用者認證的IAM工作〉。

聯合身分

最佳作法是要求人類使用者 (包括需要系統管理員存取權的使用者) 使用與身分識別提供者的同盟，才能使用臨時認證 AWS 服務 來存取。

聯合身分識別是來自企業使用者目錄的使用者、Web 身分識別提供者、Identi ty Center 目錄，或使用透過身分識別來源提供的認證進行存取 AWS 服務 的任何使用者。 AWS Directory Service同盟身分存取時 AWS 帳戶，他們會假設角色，而角色則提供臨時認證。

對於集中式存取權管理，我們建議您使用 AWS IAM Identity Center。您可以在 IAM Identity Center 中建立使用者和群組，也可以連線並同步至您自己身分識別來源中的一組使用者和群組，以便在所有應用程式 AWS 帳戶 和應用程式中使用。如需IAM身分識別中心的相關資訊，請參閱IAM識別中心是什麼？在《AWS IAM Identity Center 使用者指南》中。

IAM 使用者和群組

IAM使用者是您內部的身分，具 AWS 帳戶 有單一人員或應用程式的特定權限。在可能的情況下，我們建議您仰賴臨時登入資料，而不要建立具有長期認證 (例如密碼和存取金鑰) 的IAM使用者。不過，如果您的特定使用案例需要使用IAM者的長期認證，建議您輪換存取金鑰。如需詳細資訊，請參閱《使用指南》中的「IAM定期輪換存取金鑰」以瞭解需要長期認證的使用案例。

IAM群組是指定IAM使用者集合的身分識別。您無法以群組身分簽署。您可以使用群組來一次為多名使用者指定許可。群組可讓管理大量使用者許可的程序變得更為容易。例如，您可以擁有一個名為的群組，IAMAdmins並授與該群組管理IAM資源的權限。

使用者與角色不同。使用者只會與單一人員或應用程式建立關聯，但角色的目的是在由任何需要它的人員取得。使用者擁有永久的長期憑證，但角色僅提供暫時憑證。要了解更多信息，請參閱《IAM用戶指南》中的創建用戶（而不是角色）的IAM時間。

IAM角色

IAM角色是您 AWS 帳戶 中具有特定權限的身份。它類似於用IAM戶，但不與特定人員相關聯。您可以 AWS Management Console 透過切換角色來暫時擔任中的角色。IAM您可以透過呼叫 AWS CLI 或 AWS API作業或使用自訂來擔任角色URL。如需有關使用角色方法的詳細資訊，請參閱《使用指南》中的IAM〈使用IAM角色〉。

IAM具有臨時認證的角色在下列情況下很有用：

- 聯合身分使用者存取 — 如需向聯合身分指派許可，請建立角色，並為角色定義許可。當聯合身分進行身分驗證時，該身分會與角色建立關聯，並獲授予由角色定義的許可。如需聯合角色的相關資訊，請參閱《使用指南》中的〈建立第三方身分識別提供IAM者的角色〉。如果您使用IAM身分識別中心，則需要設定權限集。為了控制身分驗證後可以存取的內IAM容，IAMIdentity Center 會將權限集與中的角色相關聯。如需有關許可集的資訊，請參閱 AWS IAM Identity Center 使用者指南中的許可集。

- 暫時IAM使用者權限 — IAM 使用者或角色可以假定某個IAM角色，暫時取得特定工作的不同權限。
- 跨帳戶存取 — 您可以使用IAM角色允許不同帳戶中的某個人 (受信任的主體) 存取您帳戶中的資源。角色是授予跨帳戶存取權的主要方式。但是，對於某些策略 AWS 服務，您可以將策略直接附加到資源（而不是使用角色作為代理）。若要瞭解跨帳戶存取角色與以資源為基礎的政策之間的差異，請參閱《IAM使用指南》IAM中的〈跨帳號資源存取〉。
- 跨服務訪問 — 有些 AWS 服務 使用其他 AWS 服務功能。例如，當您在服務中撥打電話時，該服務通常會在 Amazon 中執行應用程式EC2或將物件存放在 Amazon S3 中。服務可能會使用呼叫主體的許可、使用服務角色或使用服務連結角色來執行此作業。
  - 轉寄存取工作階段 (FAS) — 當您使用使用IAM者或角色執行中的動作時 AWS，您會被視為主參與者。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS會使用主參與者呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。FAS只有當服務收到需要與其他 AWS 服務 資源互動才能完成的請求時，才會發出請求。在此情況下，您必須具有執行這兩個動作的許可。有關提出FAS請求時的策略詳細信息，請參閱轉發訪問會話。
  - 服務角色 — 服務角色是指服務代表您執行動作所代表的IAM角色。IAM管理員可以從中建立、修改和刪除服務角色IAM。如需詳細資訊，請參閱《IAM使用指南》 AWS 服務中的建立角色以將權限委派給
  - 服務連結角色 — 服務連結角色是連結至. AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中， AWS 帳戶 且屬於服務所有。IAM管理員可以檢視 (但無法編輯服務連結角色) 的權限。
- 在 Amazon 上執行的應用程式 EC2 — 您可以使用IAM角色來管理在執行個體上EC2執行的應用程式以及發出 AWS CLI 或 AWS API請求的臨時登入資料。這比在EC2實例中存儲訪問密鑰更好。若要將 AWS 角色指派給EC2執行個體並讓其所有應用程式都能使用，請建立附加至執行個體的執行個體設定檔。執行個體設定檔包含角色，可讓執行個體上EC2執行的程式取得臨時登入資料。如需詳細資訊，請參閱使用者指南中的使用IAM角色將許可授與在 Amazon EC2 執行個體上執行的應IAM用程式。

要了解是否使用IAM角色還是用IAM戶，請參閱《用戶指南》中的「IAM創建IAM角色的時機（而不是用戶）」。

使用政策管理存取權

您可以透 AWS 過建立原則並將其附加至 AWS 身分識別或資源來控制中的存取。原則是一個物件 AWS ，當與身分識別或資源相關聯時，會定義其權限。 AWS 當主參與者 (使用者、root 使用者或角色工作階段) 提出要求時，評估這些原則。政策中的許可決定是否允許或拒絕請求。大多數原則會 AWS 以JSON文件的形式儲存在中。如需有關JSON原則文件結構和內容的詳細資訊，請參閱《IAM使用指南》中的策略概觀。JSON

管理員可以使用 AWS JSON策略來指定誰可以存取什麼內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

預設情況下，使用者和角色沒有許可。若要授與使用者對所需資源執行動作的權限，IAM管理員可以建立IAM策略。然後，系統管理員可以將IAM原則新增至角色，使用者可以擔任這些角色。

IAM原則會定義動作的權限，不論您用來執行作業的方法為何。例如，假設您有一個允許 `iam:GetRole` 動作的政策。具有該原則的使用者可以從 AWS Management Console AWS CLI、或取得角色資訊 AWS API。

身分型政策

以身分識別為基礎的原則是您可以附加至身分識別 (例如使用者、使用IAM者群組或角色) 的JSON權限原則文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。若要瞭解如何建立以身分識別為基礎的策略，請參閱IAM使用指南中的建立IAM策略。

身分型政策可進一步分類成內嵌政策或受管政策。內嵌政策會直接內嵌到單一使用者、群組或角色。受管理的策略是獨立策略，您可以將其附加到您的 AWS 帳戶. 受管政策包括 AWS 受管政策和客戶管理的策略。若要了解如何在受管策略或內嵌策略之間進行選擇，請參閱《IAM使用手冊》中的「在受管策略和內嵌策略之間進行選擇」。

資源型政策

以資源為基礎的JSON策略是您附加至資源的政策文件。以資源為基礎的政策範例包括IAM角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條件下執行的動作。您必須在資源型政策中指定主體。主參與者可以包括帳戶、使用者、角色、同盟使用者或。 AWS 服務

資源型政策是位於該服務中的內嵌政策。您無法在以資源為基礎的策略IAM中使用 AWS 受管政策。

存取控制清單 (ACLs)

存取控制清單 (ACLs) 控制哪些主參與者 (帳戶成員、使用者或角色) 具有存取資源的權限。ACLs類似於以資源為基礎的策略，雖然它們不使用JSON政策文件格式。

Amazon S3 和 Amazon VPC 是支持服務的示例ACLs。 AWS WAF若要進一步了解ACLs，請參閱 Amazon 簡單儲存服務開發人員指南中的存取控制清單 (ACL) 概觀。

其他政策類型

AWS 支援其他較不常見的原則類型。這些政策類型可設定較常見政策類型授予您的最大許可。

- 權限界限 — 權限界限是一項進階功能，您可以在其中設定以身分識別為基礎的原則可授與給IAM實體 (IAM使用者或角色) 的最大權限。您可以為實體設定許可界限。所產生的許可會是實體的身分型政策和其許可界限的交集。會在 `Principal` 欄位中指定使用者或角色的資源型政策則不會受到許可界限限制。所有這類政策中的明確拒絕都會覆寫該允許。如需有關權限界限的詳細資訊，請參閱《IAM使用指南》中的IAM實體的權限界限。
- 服務控制策略 (SCPs) — SCPs 是指定中組織或組織單位 (OU) 最大權限的JSON策略 AWS Organizations。 AWS Organizations 是一種用於分組和集中管理您企業擁 AWS 帳戶 有的多個服務。如果您啟用組織中的所有功能，則可以將服務控制策略 (SCPs) 套用至您的任何或所有帳戶。SCP限制成員帳戶中實體的權限，包括每個帳戶 AWS 帳戶根使用者。如需有關 Organizations 的詳細資訊SCPs，請參閱AWS Organizations 使用指南中的服務控制原則。
- 工作階段政策 – 工作階段政策是一種進階政策，您可以在透過編寫程式的方式建立角色或聯合使用者的暫時工作階段時，作為參數傳遞。所產生工作階段的許可會是使用者或角色的身分型政策和工作階段政策的交集。許可也可以來自資源型政策。所有這類政策中的明確拒絕都會覆寫該允許。如需詳細資訊，請參閱《IAM使用指南》中的工作階段原則。

多種政策類型

將多種政策類型套用到請求時，其結果形成的許可會更為複雜、更加難以理解。若要瞭解如何在涉及多個原則類型時 AWS 決定是否允許要求，請參閱IAM使用指南中的原則評估邏輯。

AWS 平行運算服務如何搭配使用 IAM

在您用IAM來管理存取權之前 AWS PCS，請先瞭解哪些IAM功能可搭配使用 AWS PCS。

IAM可與 AWS 平行運算服務搭配使用的功能

IAM特徵	AWS PCS支持
身分型政策	是
資源型政策	否
政策動作	是
政策資源	是
政策條件索引鍵 (服務特定)	是

IAM特徵	AWS PCS支持
ACLs	否
ABAC(策略中的標籤)	是
暫時性憑證	是
主體許可	是
服務角色	否
服務連結角色	是

若要深入瞭解其他 AWS 服務如何 AWS PCS與大部分IAM功能搭配使用，請參閱IAM使用者指南IAM中的使用AWS 服務。

以身分識別為基礎的原則 AWS PCS

支援身分型政策：是

以身分識別為基礎的原則是您可以附加至身分識別 (例如使用者、使用IAM者群組或角色) 的JSON權限原則文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。若要瞭解如何建立以身分識別為基礎的策略，請參閱IAM使用指南中的建立IAM策略。

使用以IAM身分識別為基礎的策略，您可以指定允許或拒絕的動作和資源，以及允許或拒絕動作的條件。您無法在身分型政策中指定主體，因為這會套用至連接的使用者或角色。若要瞭解可在JSON策略中使用的所有元素，請參閱《使用IAM者指南》中的IAMJSON策略元素參考資料。

以身分識別為基礎的原則範例 AWS PCS

若要檢視以 AWS PCS身分為基礎的原則範例，請參閱。AWS 平行運算服務的身分識別原則範例

以資源為基礎的政策 AWS PCS

支援資源型政策：否

以資源為基礎的JSON策略是您附加至資源的政策文件。以資源為基礎的政策範例包括IAM角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條

件下執行的動作。您必須在資源型政策中指定主體。主參與者可以包括帳戶、使用者、角色、同盟使用者或。 AWS 服務

若要啟用跨帳戶存取，您可以在以資源為基礎的策略中指定一個或多個帳戶中的一個或多個帳戶中的IAM實體作為主體。新增跨帳戶主體至資源型政策，只是建立信任關係的一半。當主參與者和資源不同時 AWS 帳戶，受信任帳戶中的IAM管理員也必須授與主參與者實體 (使用者或角色) 權限，才能存取資源。其透過將身分型政策連接到實體來授與許可。不過，如果資源型政策會為相同帳戶中的主體授予存取，這時就不需要額外的身分型政策。如需詳細資訊，請參閱《IAM使用指南》IAM中的〈跨帳號資源存取〉。

的政策動作 AWS PCS

支援政策動作：是

管理員可以使用 AWS JSON策略來指定誰可以存取什麼內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

JSON策略Action元素描述了您可以用來允許或拒絕策略中存取的動作。策略動作通常與關聯的 AWS API操作具有相同的名稱。有一些例外情況，例如沒有匹配API操作的僅限權限的操作。也有一些作業需要政策中的多個動作。這些額外的動作稱為相依動作。

政策會使用動作來授予執行相關聯動作的許可。

若要查看 AWS PCS動作清單，請參閱服務授權參考資料中的 AWS 平行運算服務定義的動作。

中的策略動作在動作之前 AWS PCS使用下列前置詞：

```
pcs
```

若要在單一陳述式中指定多個動作，請用逗號分隔。

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

若要檢視以 AWS PCS身分為基礎的原則範例，請參閱。AWS 平行運算服務的身分識別原則範例

的政策資源 AWS PCS

支援政策資源：是

管理員可以使用 AWS JSON策略來指定誰可以存取什麼內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

ResourceJSON原則元素會指定要套用動作的一個或多個物件。陳述式必須包含 Resource 或 NotResource 元素。最佳做法是使用其 Amazon 資源名稱 (ARN) 指定資源。您可以針對支援特定資源類型的動作 (稱為資源層級許可) 來這麼做。

對於不支援資源層級許可的動作 (例如列出操作)，請使用萬用字元 (*) 來表示陳述式適用於所有資源。

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

若要查看 AWS PCS資源類型及其清單ARNs，請參閱服務授權參考資料中的 AWS 平行運算服務定義的資源。若要瞭解您可以針對每個資源指定哪些動作，請參閱 AWS 平行運算服務定義ARN的動作。

若要檢視以 AWS PCS身分為基礎的原則範例，請參閱。AWS 平行運算服務的身分識別原則範例

的政策條件索引鍵 AWS PCS

支援服務特定政策條件金鑰：是

管理員可以使用 AWS JSON策略來指定誰可以存取什麼內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

Condition 元素 (或 Condition 區塊) 可讓您指定使陳述式生效的條件。Condition 元素是選用項目。您可以建立使用條件運算子的條件運算式 (例如等於或小於)，來比對政策中的條件和請求中的值。

若您在陳述式中指定多個 Condition 元素，或是在單一 Condition 元素中指定多個索引鍵， AWS 會使用邏輯 AND 操作評估他們。如果您為單一條件索引鍵指定多個值，請使用邏輯OR運算來 AWS 評估條件。必須符合所有條件，才會授與陳述式的許可。

您也可以在指定條件時使用預留位置變數。例如，只有在IAM使用者名稱標記資源時，您才可以授與IAM使用者存取資源的權限。如需詳細資訊，請參閱《IAM使用指南》中的IAM政策元素：變數和標籤。

AWS 支援全域條件金鑰和服務特定條件金鑰。若要查看所有 AWS 全域條件索引鍵，請參閱《使用指南》中的AWS 全域條件內IAM容索引鍵。

若要查看 AWS PCS條件索引鍵清單，請參閱服務授權參考資料中的AWS 平行運算服務的條件索引鍵。若要瞭解可以使用條件索引鍵的動作和資源，請參閱AWS 平行運算服務定義的動作。

若要檢視以 AWS PCS身分為基礎的原則範例，請參閱。AWS 平行運算服務的身分識別原則範例

ACLs在 AWS PCS

支持ACLs：無

存取控制清單 (ACLs) 控制哪些主參與者 (帳戶成員、使用者或角色) 具有存取資源的權限。ACLs類似於以資源為基礎的策略，雖然它們不使用JSON政策文件格式。

ABAC與 AWS PCS

支援 ABAC (策略中的標籤): 是

以屬性為基礎的存取控制 (ABAC) 是一種授權策略，可根據屬性定義權限。在中 AWS，這些屬性稱為標籤。您可以將標籤附加至IAM實體 (使用者或角色) 和許多 AWS 資源。標記實體和資源是的第一步ABAC。然後，您可以設計ABAC策略，以便在主參與者的標籤與他們嘗試存取的資源上的標籤相符時允許作業。

ABAC在快速成長的環境中很有幫助，並且有助於原則管理變得繁瑣的情況。

如需根據標籤控制存取，請使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 條件索引鍵，在政策的條件元素中，提供標籤資訊。

如果服務支援每個資源類型的全部三個條件金鑰，則對該服務而言，值為 Yes。如果服務僅支援某些資源類型的全部三個條件金鑰，則值為 Partial。

如需有關的詳細資訊ABAC，請參閱什麼是ABAC？ 在《IAM使用者指南》中。若要檢視包含設定步驟的自學課程ABAC，請參閱《使用指南》中的〈使用以屬性為基礎的存取控制 (ABAC) IAM 〉。

使用臨時登入資料 AWS PCS

支援臨時憑證：是

當您使用臨時憑據登錄時，有些 AWS 服務 不起作用。如需其他資訊，包括哪些 AWS 服務 與臨時登入資料搭配使用 AWS 服務 ，請參閱《IAM使用指南》IAM中的使用方式。

如果您使用除了使用者名稱和密碼以外的任何方法登入，則您正在 AWS Management Console 使用臨時認證。例如，當您 AWS 使用公司的單一登入 (SSO) 連結存取時，該程序會自動建立臨時認證。當您以使用者身分登入主控台，然後切換角色時，也會自動建立臨時憑證。如需有關切換角色的詳細資訊，請參閱《IAM使用者指南》中的〈切換到角色 (主控台)〉。

您可以使用 AWS CLI 或手動建立臨時認證 AWS API。然後，您可以使用這些臨時登入資料來存取 AWS。 AWS 建議您動態產生臨時登入資料，而不是使用長期存取金鑰。如需詳細資訊，請參閱IAM。

的跨服務主體權限 AWS PCS

支援轉寄存取工作階段 (FAS)：是

當您使用使用IAM者或角色在中執行動作時 AWS，您會被視為主參與者。使用某些服務時，您可能會執行某個動作，進而在不同服務中啟動另一個動作。FAS會使用主參與者呼叫的權限 AWS 服務，並結合要求 AWS 服務 向下游服務發出要求。FAS只有當服務收到需要與其他 AWS 服務 資源互動才能完成的請求時，才會發出請求。在此情況下，您必須具有執行這兩個動作的許可。有關提出FAS請求時的策略詳細信息，請參閱轉發訪問會話。

AWS PCS 的服務角色

支援服務角色：否

服務角色是服務假定代表您執行動作的IAM角色。IAM管理員可以從中建立、修改和刪除服務角色 IAM。如需詳細資訊，請參閱《IAM使用指南》 AWS 服務中的建立角色以將權限委派給

> **Warning**
>
> 變更服務角色的權限可能會中斷 AWS PCS功能。只有在 AWS PCS提供指引時才編輯服務角色。

服務連結角色 AWS PCS

支援服務連結角色：是

服務連結角色是一種連結至. AWS 服務服務可以擔任代表您執行動作的角色。服務連結角色會顯示在您的中， AWS 帳戶 且屬於服務所有。IAM管理員可以檢視 (但無法編輯服務連結角色) 的權限。

如需有關建立或管理服務連結角色的詳細資訊，請參閱使用IAM的AWS 服務。在表格中尋找服務，其中包含服務連結角色欄中的 Yes。選擇是連結，以檢視該服務的服務連結角色文件。

AWS 平行運算服務的身分識別原則範例

依預設，使用者和角色沒有建立或修改 AWS PCS資源的權限。他們也無法使用 AWS Management Console、 AWS Command Line Interface (AWS CLI) 或執行工作 AWS API。若要授與使用者對所需

資源執行動作的權限，IAM管理員可以建立IAM策略。然後，系統管理員可以將IAM原則新增至角色，使用者可以擔任這些角色。

若要瞭解如何使用這些範例原則文件來建立以IAM身分識別為基礎的JSON策略，請參閱使用指南中的IAM建立IAM策略。

如需有關由所定義之動作和資源類型的詳細資訊 AWS PCS，包括每個ARNs資源類型的格式，請參閱服務授權參考中的 AWS 平行運算服務的動作、資源和條件索引鍵。

主題

- 政策最佳實務
- 使用 AWS PCS 主控台
- 允許使用者檢視他們自己的許可

政策最佳實務

以身分識別為基礎的政策會決定某人是否可以建立、存取或刪除您帳戶中的 AWS PCS資源。這些動作可能會讓您的 AWS 帳戶產生費用。當您建立或編輯身分型政策時，請遵循下列準則及建議事項：

- 開始使用 AWS 受管原則並邁向最低權限權限 — 若要開始授與使用者和工作負載的權限，請使用可授與許多常見使用案例權限的AWS 受管理原則。它們可用在您的 AWS 帳戶. 建議您透過定義特定於您使用案例的 AWS 客戶管理政策，進一步降低使用權限。如需詳細資訊，請參閱AWS 《IAM使用指南》中針對工作職能的AWS 受管理策略或受管理的策略。
- 套用最低權限權限 — 當您使用原則設定權限時，IAM只授與執行工作所需的權限。為實現此目的，您可以定義在特定條件下可以對特定資源採取的動作，這也稱為最低權限許可。如需有關使用套用權限IAM的詳細資訊，請參閱《使用指南》IAM中的IAM《策略與權限》。
- 使用IAM策略中的條件進一步限制存取 — 您可以在策略中新增條件，以限制對動作和資源的存取。例如，您可以撰寫政策條件，以指定必須使用傳送所有要求SSL。您也可以使用條件來授與對服務動作的存取權 (如透過特定) 使用這些動作 AWS 服務，例如 AWS CloudFormation。如需詳細資訊，請參閱《IAM使用指南》中的IAMJSON策略元素：條件。
- 使用 IAM Access Analyzer 驗證您的原IAM則，以確保安全和功能性的權限 — IAM Access Analyzer會驗證新的和現有的原則，以便原則遵循IAM原則語言 (JSON) 和IAM最佳做法。IAMAccess Analyzer 提供超過 100 項原則檢查和可行的建議，協助您撰寫安全且功能正常的原則。如需詳細資訊，請參閱IAM使IAM用指南中的存取分析器原則驗證。
- 需要多因素驗證 (MFA) — 如果您的案例需要使IAM用者或 root 使用者 AWS 帳戶，請開啟以取得額外MFA的安全性。若要在呼叫API作業MFA時需要，請在原則中新增MFA條件。如需詳細資訊，請參閱《IAM使用指南》中的 < 設定MFA受保護的API存取 > 。

如需中最佳作法的詳細資訊IAM，請參閱《IAM使用指南》IAM中的「安全性最佳作法」。

使用 AWS PCS 主控台

若要存取「 AWS 平行運算服務」主控台，您必須擁有最少一組權限。這些權限必須允許您列出和檢視有關 AWS 帳戶. AWS PCS 如果您建立比最基本必要許可更嚴格的身分型政策，則對於具有該政策的實體 (使用者或角色) 而言，主控台就無法如預期運作。

您不需要為只對 AWS CLI 或撥打電話的使用者允許最低主控台權限 AWS API。相反地，只允許存取符合他們嘗試執行之API作業的動作。

如需使用主控台所需的最低權限的詳細資 AWS PCS訊，請參閱的最低權限 AWS PCS。

允許使用者檢視他們自己的許可

此範例顯示如何建立原則，讓使IAM用者檢視附加至其使用者身分識別的內嵌和受管理原則。此原則包含在主控台上或以程式設計方式使用或完成此動作的 AWS CLI 權限 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": "*"
        }
    ]
}
```

AWSAWS 平行運算服務的管理原則

受 AWS 管理的策略是由建立和管理的獨立策略 AWS。 AWS 受管理的策略旨在為許多常見使用案例提供權限，以便您可以開始將權限指派給使用者、群組和角色。

請記住， AWS 受管理的政策可能不會為您的特定使用案例授與最低權限權限，因為這些權限可供所有 AWS 客戶使用。我們建議您定義使用案例專屬的客戶管理政策，以便進一步減少許可。

您無法變更受 AWS 管理策略中定義的權限。如果 AWS 更新 AWS 受管理原則中定義的權限，則此更新會影響附加原則的所有主體識別 (使用者、群組和角色)。 AWS 當新的啟動或新API作業可供現有服務使 AWS 服務 用時，最有可能更新 AWS 受管理的策略。

如需詳細資訊，請參閱IAM使用指南中的AWS 受管理策略。

AWS 受管理的策略： AWSPCSServiceRolePolicy

您無法附加 AWSPCSServiceRolePolicy 至您的IAM實體。此原則附加至服務連結角色，可 AWS PCS 代表您執行動作。如需詳細資訊，請參閱AWS PCS 的服務連結角色。

許可詳細資訊

此政策包含以下許可。

- `ec2`— 允許 AWS PCS創建和管理 Amazon EC2 資源。
- `iam`— 允許 AWS PCS為 Amazon EC2 車隊創建服務鏈接角色並將角色傳遞給 Amazon EC2。
- `cloudwatch`— 允許 AWS PCS將服務指標發佈到 Amazon CloudWatch。
- `secretsmanager`— 允許管 AWS PCS理 AWS PCS叢集資源的密碼。

```
{
    "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 PCS AWS 受管理策略的更新

檢視 AWS PCS自此服務開始追蹤這些變更以來的 AWS 受管理策略更新詳細資料。如需有關此頁面變更的自動警示，請訂閱「 AWS PCS文件記錄」頁面上的RSS摘要。

變更	描述	日期
AWS PCS 已開始追蹤變更	AWS PCS開始追蹤其 AWS 受管理策略的變更。	2024年8月28日

AWS PCS 的服務連結角色

AWS 平行運算服務使用 AWS Identity and Access Management (IAM) 服務連結角色。服務連結角色是直接連結至 AWS PCS的唯一IAM角色類型。服務連結角色由預先定義， AWS PCS並包含服務代表您呼叫其他服 AWS 務所需的所有權限。

服務連結角色可讓您 AWS PCS更輕鬆地設定，因為您不需要手動新增必要的權限。 AWS PCS定義其服務連結角色的權限，除非另有定義，否則只 AWS PCS能擔任其角色。定義的權限包括信任原則和權限原則，而且該權限原則無法附加至任何其他IAM實體。

您必須先刪除角色的相關資源，才能刪除服務連結角色。這樣可以保護您的 AWS PCS資源，因為您不會意外移除存取資源的權限。

如需支援服務連結角色之其他服務的相關資訊，請參閱使用的AWS 服務，IAM並在服務連結角色欄中尋找具有是的服務。選擇具有連結的是，以檢視該服務的服務連結角色文件。

服務連結角色權限 AWS PCS

AWS PCS使用名為的服務連結角色 AWSServiceRoleForPCS— AWS PCS 允許管理 Amazon EC2 資源。

服 AWSServiceRoleForPCS 務連結角色會信任下列服務擔任該角色：

- `pcs.amazonaws.com`

名為的角色權限原則AWSPCSServiceRolePolicy允許 AWS PCS對特定資源完成動作。

您必須設定許可，以允許您的使用者、群組或角色建立、編輯或刪除服務連結角色。如需詳細資訊，請參閱IAM使用指南中的服務連結角色權限。

建立服務連結角色 AWS PCS

您不需要手動建立服務連結角色。 AWS PCS建立叢集時，會為您建立服務連結角色。

編輯下列項目的服務連結角色 AWS PCS

AWS PCS不允許您編輯 AWSServiceRoleForPCS 服務連結角色。因為有各種實體可能會參考服務連結角色，所以您無法在建立角色之後變更角色名稱。但是，您可以使用編輯角色的描述IAM。如需詳細資訊，請參閱IAM使用指南中的編輯服務連結角色。

刪除下列項目的服務連結角色 AWS PCS

若您不再使用需要服務連結角色的功能或服務，我們建議您刪除該角色。如此一來，您就沒有未主動監控或維護的未使用實體。然而，在手動刪除服務連結角色之前，您必須先清除資源。

> **Note**
>
> 當您嘗試刪除資源時，如果 AWS PCS服務正在使用此角色，則刪除可能會失敗。若此情況發生，請等待數分鐘後並再次嘗試操作。

若要移除使用的 AWS PCS資源 AWSServiceRoleForPCS

您必須刪除所有叢集，才能刪除 AWSServiceRoleForPCS 服務連結角色。如需詳細資訊，請參閱刪除叢集。

若要使用手動刪除服務連結角色 IAM

使用IAM主控台 AWS CLI、或刪除 AWSServiceRoleForPCS 服務連結角色。 AWS API如需詳細資訊，請參閱IAM使用指南中的刪除服務連結角色。

AWS PCS 服務連結角色的支援區域

AWS PCS支援在所有提供服務的區域中使用服務連結角色。如需詳細資訊，請參閱AWS 區域與端點。

Amazon EC2 現貨角色 AWS PCS

如果您想要建立使用 Spot 作為其購買選項的 AWS PCS計算節點群組，您 AWS 帳戶也必須在. AWSServiceRoleForEC2Spot 您可以使用以下 AWS CLI 命令來創建角色。如需詳細資訊，請參閱AWS Identity and Access Management 使用指南中的建立服務連結角色和建立角色以將權限委派給 AWS 服務。

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

> **Note**
>
> 如果您 AWS 帳戶 已經擁有AWSServiceRoleForEC2SpotIAM角色，您會收到下列錯誤訊息。

```
 An error occurred (InvalidInput) when calling the CreateServiceLinkedRole
operation: Service role name AWSServiceRoleForEC2Spot has been taken in this
account, please try a different suffix.
```

的最低權限 AWS PCS

本節說明身分IAM識別 (使用者、群組或角色) 使用服務所需的最低IAM權限。

內容

使用API動作的最低權限

API動作	最低許可	控制台的其他權限
CreateCluster	`ec2:CreateNetworkInterface, ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:GetSecurityGroupsForVpc, iam:CreateServiceLinkedRole, secretsmanager:CreateSecret, secretsmanager:TagResource, pcs:CreateCluster`	
ListClusters	`pcs:ListClusters`	

API動作	最低許可	控制台的其他權限
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`

API動作	最低許可	控制台的其他權限
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`

API動作	最低許可	控制台的其他權限
DeleteQueue	`pcs:DeleteQueue`	

使用標籤所需的最低權限

在中使用標籤時，需要下列權限 AWS PCS。

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

支援記錄檔所需的最低權限

AWS PCS將日誌資料傳送到 Amazon CloudWatch 日誌 (CloudWatch 日誌)。您必須確保您的身份具有使用 CloudWatch 日誌的最低權限。如需詳細資訊，請參閱 Amazon CloudWatch 日誌使用者指南中的管理 CloudWatch 日誌資源存取許可概觀。

如需將日誌傳送到日誌的服務所需許可的詳細資訊，請參閱 Amazon CloudWatch Lo CloudWatch gs 使用者指南中的啟用 AWS 服務記錄。

服務管理員的最低權限

下列IAM原則指定IAM身分識別 (使用者、群組或角色) 設定及管理 AWS PCS服務所需的最低權限。

> **Note**
>
> 不設定和管理服務的使用者不需要這些權限。只執行工作的使用者會使用安全殼層 (SSH) 來連線至叢集。 AWS Identity and Access Management （ IAM ）不處理的身份驗證或授權SSH。

```
{
    "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": "*"
        }
    ]
}
```

您可以從策略中排除下列權限，而是在中使用對應的受管理策略IAM：

- "firehose:*"

  ```
  AmazonKinesisFirehoseFullAccess
  ```

- "kms:*"

  ```
  AWSKeyManagementServicePowerUser
  ```

- "logs:*"

  ```
  CloudWatchLogsFullAccess
  ```

- "s3:*"

  ```
  AmazonS3FullAccess
  ```

IAM AWS 平行運算服務執行個體設定檔

在執行個EC2體上執行的應用程式必須在其發出的任何 AWS API要求中包含 AWS 認證。我們建議您使用IAM角色來管理EC2執行個體上的臨時登入資料。您可以定義執行個體設定檔來執行此操作，並將其附加至您的執行個體。如需詳細資訊，請參閱 Amazon 彈性運算雲端使用者指南EC2中的 Amazon IAM 角色。

> **Note**
>
> 使用為 Amazon 建立IAM角色時EC2，主控台會自動建立執行個體設定檔，並為其指定與IAM角色相同的名稱。 AWS Management Console 如果您使用 AWS CLI、動 AWS API作或建立IAM角色，您可 AWS SDK以將執行個體設定檔建立為單獨的動作。如需詳細資訊，請參閱 Amazon 彈性運算雲端使用者指南中的執行個體設定檔。

建立運算節點群組時，必須指定執行個體設定檔ARN的。您可以為部分或所有運算節點群組選擇不同的執行個體設定檔。

例項設定檔需求

例項設定檔名稱

IAM執行個體設定檔ARN必須以其路徑開頭AWSPCS或包含/aws-pcs/。

Example

- `arn:aws:iam::*:instance-profile/AWSPCS-example-role-1` 和
- `arn:aws:iam::*:instance-profile/aws-pcs/example-role-2`.

許可

的執行個體設定檔至少 AWS PCS必須包含下列政策。它可讓運算節點在運作時通知 AWS PCS服務。

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

```
            "Effect": "Allow"
        }
    ]
}
```

其他政策

您可以考慮將受管政策新增至執行個體設定檔。例如：

- 亞馬遜 S3 ReadOnlyAccess 提供對所有 S3 儲存貯體的唯讀存取。
- mazonSSMManagedInstanceCore啟用 AWS Systems Manager 服務核心功能，例如直接從 Amazon 管理主控台進行遠端存取。
- CloudWatchAgentServerPolicy包含 AmazonCloudWatchAgent 在伺服器上使用所需的權限。

您也可以加入自己的IAM原則，以支援您的特定使用案例。

建立執行個體描述檔

您可以直接從 Amazon EC2 主控台建立執行個體設定檔。如需詳細資訊，請參閱使用指南中的AWS Identity and Access Management 使用執行個體設定檔。

疑難排解 AWS 平行運算服務識別與存取

使用下列資訊可協助您診斷及修正使用和時可能會遇到的 AWS PCS常見問題IAM。

主題

- 我沒有執行操作的授權 AWS PCS
- 我沒有授權執行 iam：PassRole
- 我想允許我以外的人訪 AWS 帳戶 問我的 AWS PCS資源

我沒有執行操作的授權 AWS PCS

如果您收到錯誤，告知您未獲授權執行動作，您的政策必須更新，允許您執行動作。

當使用mateojacksonIAM者嘗試使用主控台來檢視虛構*my-example-widget*資源的詳細資料，但沒有虛構的pcs:*GetWidget*權限時，就會發生下列範例錯誤。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 pcs:GetWidget on resource: my-example-widget
```

在此情況下，必須更新 mateojackson 使用者的政策，允許使用 pcs:*GetWidget* 動作存取 *my-example-widget* 資源。

如果您需要協助，請聯絡您的 AWS 系統管理員。您的管理員提供您的簽署憑證。

我沒有授權執行 iam：PassRole

如果您收到未獲授權執行iam:PassRole動作的錯誤訊息，則必須更新您的原則以允許您將角色傳遞給 AWS PCS。

有些 AWS 服務 允許您將現有角色傳遞給該服務，而不是建立新的服務角色或服務連結角色。如需執行此作業，您必須擁有將角色傳遞至該服務的許可。

當名為的使用IAM者marymajor嘗試使用主控台執行中的動作時，就會發生下列範例錯誤 AWS PCS。但是，動作請求服務具備服務角色授予的許可。Mary 沒有將角色傳遞至該服務的許可。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

在這種情況下，Mary 的政策必須更新，允許她執行 iam:PassRole 動作。

如果您需要協助，請聯絡您的 AWS 系統管理員。您的管理員提供您的簽署憑證。

我想允許我以外的人訪 AWS 帳戶 問我的 AWS PCS資源

您可以建立一個角色，讓其他帳戶中的使用者或您組織外部的人員存取您的資源。您可以指定要允許哪些信任物件取得該角色。對於支援以資源為基礎的政策或存取控制清單 (ACLs) 的服務，您可以使用這些政策授與人員存取您的資源。

如需進一步了解，請參閱以下內容：

- 若要瞭解是否 AWS PCS支援這些功能，請參閱AWS 平行運算服務如何搭配使用 IAM。
- 若要瞭解如何提供您所擁有資源 AWS 帳戶 的存取權，請參閱《IAM使用者指南》中 AWS 帳戶 的〈提供存取權給其他IAM使用者〉。
- 若要瞭解如何將資源存取權提供給第三方 AWS 帳戶，請參閱《IAM使用指南》中的提供第三方 AWS 帳戶 擁有的存取權。
- 若要瞭解如何透過身分聯盟提供存取權，請參閱使用指南中的提供對外部驗證使用IAM者的存取權(身分聯合)。

- 若要瞭解針對跨帳號存取使用角色與以資源為基礎的政策之間的差異，請參閱《使用IAM者指南》IAM中的〈跨帳號資源存取〉。

AWS 平行運算服務的合規性驗證

若要瞭解 AWS 服務 是否屬於特定規範遵循方案的範圍內，請參閱AWS 服務 遵循規範計劃方案中的，並選擇您感興趣的合規方案。如需一般資訊，請參閱AWS 規範計劃AWS。

您可以使用下載第三方稽核報告 AWS Artifact。如需詳細資訊，請參閱下載中的報告中的 AWS Artifact。

您在使用時的合規責任取決 AWS 服務 於資料的敏感性、公司的合規目標以及適用的法律和法規。AWS 提供下列資源以協助遵循法規：

- 安全性與合規性快速入門指南 — 這些部署指南討論架構考量，並提供部署以安全性和合規性 AWS 為重點的基準環境的步驟。
- 在 Amazon Web Services 上進行HIPAA安全與合規架構 — 本白皮書說明公司如何使用建立符合資格的應 AWS 用程HIPAA式。

> **Note**
>
> 並非所有 AWS 服務 人都HIPAA符合資格。如需詳細資訊，請參閱合HIPAA格服務參考資料。

- AWS 合規資源AWS — 此工作簿和指南集合可能適用於您的產業和所在地。
- AWS 客戶合規指南 — 透過合規的角度瞭解共同的責任模式。這份指南總結了在多個架構 (包括美國國家標準 AWS 服務 與技術研究所 (NIST)、支付卡產業安全標準委員會 () 和國際標準化組織 ()PCI) 中保護安全控制指引的最佳做法，並將其對應至安全控制。ISO
- 使用AWS Config 開發人員指南中的規則評估資源 — 此 AWS Config 服務會評估您的資源組態符合內部實務、產業準則和法規的程度。
- AWS Security Hub— 這 AWS 服務 提供了內部安全狀態的全面視圖 AWS。Security Hub 使用安全控制，可評估您的 AWS 資源並檢查您的法規遵循是否符合安全業界標準和最佳實務。如需支援的服務和控制清單，請參閱 Security Hub controls reference。
- Amazon GuardDuty — 透過監控環境中的 AWS 帳戶可疑和惡意活動，藉此 AWS 服務 偵測您的工作負載、容器和資料的潛在威脅。 GuardDuty 可協助您因應各種合規性需求 PCIDSS，例如符合特定合規性架構所要求的入侵偵測需求。

- AWS Audit Manager— 這 AWS 服務 有助於您持續稽核您的 AWS 使用情況，以簡化您管理風險的方式，以及遵守法規和業界標準的方式。

AWS 平行運算服務的彈性

AWS 全球基礎架構是圍繞 AWS 區域 和可用區域建立的。 AWS 區域 提供多個實體分離和隔離的可用區域，這些區域與低延遲、高輸送量和高冗餘網路相連。透過可用區域，您可以設計與操作的應用程式和資料庫，在可用區域之間自動容錯移轉而不會發生中斷。可用區域的可用性、容錯能力和擴展能力，均較單一或多個資料中心的傳統基礎設施還高。

如需 AWS 區域 和可用區域的詳細資訊，請參閱AWS 全域基礎結構。

AWS 平行運算服務的基礎架構安全

作為一種託管服務， AWS 並行計算服務受到 AWS 全球網絡安全的保護。有關 AWS 安全服務以及如何 AWS 保護基礎結構的詳細資訊，請參閱AWS 雲端安全 若要使用基礎架構安全性的最佳做法來設計您的 AWS 環境，請參閱安全性支柱架構良 AWS 好的架構中的基礎結構保護。

您可以使用 AWS 已發佈的API呼叫透 AWS PCS過網路存取。使用者端必須支援下列專案：

- 傳輸層安全性 (TLS)。我們需要 TLS 1.2 並推薦 TLS 1.3。
- 具有完美前向保密（ ）的密碼套件，例如（短暫的迪菲-赫爾曼PFS）或DHE（橢圓曲線短暫迪菲-赫爾曼）。ECDHE現代系統(如 Java 7 和更新版本)大多會支援這些模式。

此外，請求必須使用存取金鑰 ID 和與IAM主體相關聯的秘密存取金鑰來簽署。或者，您可以透過 AWS Security Token Service (AWS STS) 來產生暫時安全憑證來簽署請求。

AWS PCS建立叢集時，服務會在服務擁有的帳戶中啟動 Slurm 控制器，與您帳戶中的運算節點分開。若要橋接控制器與運算節點之間的通訊， AWS PCS請在您VPC的. ENI Slurm 控制器使用ENI來管理不同的運算節點並與其通訊 AWS 帳戶，以維護資源的安全性和隔離，同時促進高效率HPC的 AI/ML 操作。

AWS 並行計算服務中的漏洞分析和管理

配置和 IT 控制是與您之間共同 AWS 的責任。如需詳細資訊，請參閱AWS 共用的責任模型。 AWS 處理服務帳戶中基礎結構的基本安全性工作，例如在控制器執行個體上修補作業系統、防火牆組態和基 AWS 礎結構災難復原。這些程序已由適當的第三方進行檢閱並認證。如需詳細資訊，請參閱安全性、身分識別和合規性的最佳做法。

您必須為下列項目中基礎結構的安全性負責 AWS 帳戶：

- 維護您的程式碼，包括更新和安全性修補程式。
- 修補及更新節點群組執行個體上的作業系統。
- 更新排程器以將其保留在支援的版本中。
- 驗證並加密使用者用戶端與其連線到的節點之間的通訊。

預防跨服務混淆代理人

混淆代理人問題屬於安全性問題，其中沒有執行動作許可的實體可以強制具有更多許可的實體執行該動作。在中 AWS，跨服務模擬可能會導致混淆的副問題。在某個服務 (呼叫服務) 呼叫另一個服務 (被呼叫服務) 時，可能會發生跨服務模擬。可以操縱呼叫服務來使用其許可，以其不應有存取許可的方式對其他客戶的資源採取動作。為了預防這種情況， AWS 提供的工具可協助您保護所有服務的資料，而這些服務主體已獲得您帳戶中資源的存取權。

我們建議在資源策略中使用aws:SourceArn和aws:SourceAccount全域條件內容索引鍵，以限制 AWS 平行運算服務 (AWS PCS) 為資源提供其他服務的權限。如果您想要僅允許一個資源與跨服務存取相關聯，則請使用 aws:SourceArn。如果您想要允許該帳戶中的任何資源與跨服務使用相關聯，請使用 aws:SourceAccount。

防止混淆的副問題的最有效方法是使用aws:SourceArn全局條件上下文鍵與完整ARN的資源。如果您不知道資源ARN的完整內容，或者您要指定多個資源，請針對未知部分使用萬用字元 (*) 的aws:SourceArn全域內容條件索引鍵與萬用字元 () ARN。例如：arn:aws:*servicename*:*:*123456789012*:*。

如果aws:SourceArn值不包含帳戶 ID (例如 Amazon S3 儲存貯體)ARN，您必須同時使用全域條件內容金鑰來限制許可。

的值aws:SourceArn必須是叢集ARN。

下列範例顯示如何在中使用aws:SourceArn和aws:SourceAccount全域條件前後關聯鍵字 AWS PCS來避免混淆的副問題。

```
{
"Version": "2012-10-17",
  "Statement": {
"Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
```

```
    "Principal": {
      "Service": "pcs.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": [
          "arn:aws:pcs:us-east-1:123456789012:cluster/*"
        ]
      },
      "StringEquals": {
        "aws:SourceAccount": "123456789012"
      }
    }
  }
}
```

IAM作為運算節點群組一部分佈建的 Amazon EC2 執行個體角色

AWS PCS針對叢集中的每個已設定運算節點群組自動協調 Amazon EC2 容量。建立運算節點群組時，使用者必須透過iamInstanceProfileArn欄位提供IAM執行個體設定檔。執行個體設定檔會指定與佈建的EC2執行個體相關聯的權限。 AWS PCS接受任何具有AWSPCS作為角色名稱前綴或/aws-pcs/作為角色路徑一部分的角色。建立或更新計算節點群組的IAM身分識別 (使用者或角色) 需要使用iam:PassRole權限。當使用者呼叫CreateComputeNodeGroup或動UpdateComputeNodeGroupAPI作時， AWS PCS會檢查是否允許使用者執行iam:PassRole動作。

下列範例原則會授與僅傳遞名稱開頭為之IAM角色的權限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"
                    ]
                }
```

```
            }
        }
    ]
}
```

AWS 平行運算服務的安全性最佳作法

本節說明「 AWS 平行運算服務」(AWS PCS) 特有的安全性最佳作法。若要深入了解中的安全性最佳做法 AWS，請參閱安全性、身分識別和合規性的最佳做法。

AMI相關安全

- 請勿將 AWS PCS範例用AMIs於生產工作負載。該樣本不AMIs受支持，僅用於測試。
- 定期更新 AWS PCS執行個體的作業系統和軟體，以減輕漏洞。
- 用 AWS Systems Manager 於自動修補並維持安全性原則的合規性。
- 只使用從官方 AWS 來源下載的經過驗證的官方 AWS PCS套件。
- 定期更新運算節點上的 AWS PCS套件，以接收安全性修補程式和改進。考慮將此程序自動化，以將弱點降至最低。

Surm 工作負載管理員安全性

- 實作存取控制和網路限制，以保護 Slurm 控制和運算節點。僅允許受信任的使用者和系統提交作業並存取 Slurm 管理命令。
- 使用 Slurm 的內建安全性功能，例如 Slurm 驗證，以確保工作提交和通訊都經過驗證。
- 更新 Slurm 版本以維持平穩的操作和集群支持。

> **Important**
>
> 任何使用 Slurm 版本已達到支援壽命 (EOSL) 結束的叢集都會立即停止。使用使用者指南頁面頂端的連結來訂閱 AWS PCS文件RSS摘要，以便在 Slurm 版本接近時收到通知。EOSL

監控和記錄

- 使用 Amazon CloudWatch 日誌， AWS CloudTrail 以及監控和記錄叢集和 AWS 帳戶. 使用資料進行疑難排解和稽核。

網路安全

- 單獨部署您的 AWS PCS叢集，VPC以將您的HPC環境與其他網路流量隔離開來。
- 使用安全群組和網路存取控制清單 (ACLs) 來控制 AWS PCS執行個體和子網路的輸入和輸出流量。
- 使用 AWS PrivateLink 或VPC端點來保持叢集與網路內其他 AWS 服務之間的 AWS 網路流量。

記錄和監控 AWS PCS

監控是維持其他AWS資源的可靠性、可用性和效能的 AWS PCS重要組成部分。AWS提供下列監控工具來監視 AWS PCS、在發生錯誤時回報，並在適當時自動採取行動：

- Amazon 會即時 CloudWatch監控您的 AWS 資源和執行 AWS 的應用程式。您可以收集和追蹤指標、建立自訂儀板表，以及設定警示，在特定指標達到您指定的閾值時通知您或採取動作。例如，您可以 CloudWatch 追蹤 Amazon EC2 執行個體的CPU使用情況或其他指標，並在需要時自動啟動新執行個體。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南。
- Amazon CloudWatch 日誌可讓您從 Amazon EC2 執行個體和其他來源監控 CloudTrail、存放和存取日誌檔。 CloudWatch 記錄檔可以監控記錄檔中的資訊，並在符合特定臨界值時通知您。您也可以將日誌資料存檔在高耐用性的儲存空間。如需詳細資訊，請參閱 Amazon CloudWatch 日誌使用者指南。
- AWS CloudTrail擷取您帳戶或代表您的 AWS 帳戶發出的API呼叫和相關事件，並將日誌檔傳送到您指定的 Amazon S3 儲存貯體。您可以識別呼叫的使用者和帳戶 AWS、進行呼叫的來源 IP 位址，以及呼叫發生的時間。如需詳細資訊，請參閱《AWS CloudTrail 使用者指南》https://docs.aws.amazon.com/awscloudtrail/latest/userguide/。

AWS PCS排程器記錄

您可以設定 AWS PCS為將詳細的記錄資料從叢集排程器傳送到 Amazon CloudWatch 日誌、亞馬遜簡單儲存服務 (Amazon S3) 和 Amazon 資料 Firehose。這可以協助監控和疑難排解。您可以使用 AWS PCS主控台來設定 AWS PCS排程器記錄，也可以透過程式設計方式使用 AWS CLI 或SDK.

內容

必要條件

用來管理AWSPCS叢集的IAM主體必須允許pcs:AllowVendedLogDeliveryForResource。以下是啟用此功能的範例AWSIAM原則。

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

使用 AWS PCS主控台設定排程器記錄

若要在主控台中設定AWSPCS排程器記錄檔，請依照下列步驟執行：

1. 開啟主AWS PCS控台。
2. 選擇叢集，然後瀏覽要啟用記錄日誌之 AWS PCS叢集的詳細資訊頁面。
3. 選擇 Logs (日誌)。
4. 在記錄傳送下 — 排程器記錄 — 選用
   a. 最多可新增三個記錄傳送目的地。選項包括 CloudWatch 日誌、Amazon S3 或 Firehose。
   b. 選擇 [更新記錄傳送]。

您可以重新瀏覽此頁面，以重新設定、新增或移除記錄傳送。

使用設定排程器記錄 AWS CLI

若要完成此作業，您至少需要一個傳遞目的地、一個傳遞來源 (PCS叢集) 和一個傳遞，這是將來源連接至目的地的關係。

建立傳送目的地

您至少需要一個傳遞目的地，才能從AWSPCS叢集接收排程器記錄。您可以在《 CloudWatch API使用者指南》一 PutDeliveryDestination節中進一步瞭解此主題。

若要使用建立傳送目的地 AWS CLI

- 使用下面的命令創建一個目的地。執行命令之前，請執行下列替換：
  - Replace (取代) *region-code* 與您將 AWS 區域 在那裡創建您的目的地。這通常與部署 AWS PCS叢集的區域相同。
  - Replace (取代) *pcs-logs-destination* 使用您的首選名稱。對於您帳戶中的所有送貨目的地而言，該資料必須是唯一
  - Replace (取代) *resource-arn* 使ARN用 CloudWatch 日誌中的現有日誌組，S3 存儲桶或 Firehose 中的交付流。範例包括：
    - CloudWatch 記錄檔群組

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

    - S3 bucket (S3 儲存貯體)

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

    - Firehose 投遞分流

      ```
      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
```

請記下新遞送目的地的，因ARN為您將需要它來設定傳送。

啟用 AWS PCS叢集做為傳遞來源

若要從中收集排程器記錄 AWSPCS，請將叢集設定為傳遞來源。如需詳細資訊，請參閱 Amazon CloudWatch 日誌API參考PutDeliverySource中的。

使用將叢集設定為傳遞來源 AWS CLI

- 使用以下命令啟用從叢集傳送記錄檔。執行命令之前，請執行下列替換：
  - Replace (取代) *region-code* 與您 AWS 區域 的叢集的部署位置。
  - Replace (取代) *cluster-logs-source-name* 使用此來源的名稱。它對於您中的所有傳送來源必須是唯一的 AWS 帳戶。考慮合併 AWS PCS叢集的名稱或 ID。
  - Replace (取代) *cluster-arn* 使用ARN適用於您的 AWS PCS叢集

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

將叢集傳遞來源 Connect 至傳遞目的地

若要讓排程器記錄資料從叢集流向目的地，您必須設定連線這些資料的傳遞。如需詳細資訊，請參閱 Amazon CloudWatch 日誌API參考CreateDelivery中的。

若要使用建立遞送 AWS CLI

- 使用後面的命令建立傳送。執行命令之前，請執行下列替換：
  - Replace (取代) *region-code* 與您的來源和目的 AWS 區域 地所在的位置。
  - Replace (取代) *cluster-logs-source-name* 使用上面的交付來源的名稱。
  - Replace (取代) *destination-arn* 與您想要傳送記錄檔的傳送目的地。ARN

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

排程器記錄資料流路徑和名稱

AWSPCS排程器記錄檔的路徑和名稱取決於目的地類型。

- CloudWatch 日誌
  - Lo CloudWatch gs 串流會遵循此命名慣例。

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

Example

```
AWSLogs/PCS/abcdef0123/slurmctld_24.05.log
```

- S3 bucket (S3 儲存貯體)
  - S3 儲存貯體輸出路徑遵循以下命名慣例：

```
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.
```

  - S3 物件名稱遵循以下慣例：

```
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
```

AWS PCS排程器記錄範例

AWSPCS排程器記錄是結構化的。除了 Slurm 控制器進程發出的日誌消息之外，它們還包括諸如集群標識符，調度程序類型，主要和補丁版本之類的字段。請見此處範例。

```
{
    "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"
}
```

使用 Amazon 監控 AWS 平行計算服務 CloudWatch

CloudWatch Amazon 會每隔一段時間從叢集收集指標，藉此監控您的 AWS 平行運算服務 (AWS PCS) 叢集運作狀態和效能。這些指標會保留下來，讓您存取歷史資料，並深入瞭解叢集的效能。

CloudWatch 也可讓您監控由啟動的EC2執行個體， AWS PCS以符合擴展需求。雖然您可以檢查執行中執行個體的記錄，但一旦執行個體終止，通常會刪除 CloudWatch 指標和記錄資料。不過，您可以使用EC2啟動範本在執行個體上設定 CloudWatch 代理程式，以便在執行個體終止後持續保留指標和記錄，進而進行長期監控和分析。

探索本節中的主題，以進一步瞭解 AWS PCS使用 CloudWatch.

主題

- 使用監視 AWS PCS指標 CloudWatch
- 使用 AWS PCS Amazon 監控實例 CloudWatch

使用監視 AWS PCS指標 CloudWatch

您可以使用 Amazon 監控 AWS PCS叢集運作狀態 CloudWatch，Amazon 會從叢集收集資料並將其轉換為近乎即時的指標。這些統計資料會保留 15 個月，因此您可以存取歷史資訊，並更好地瞭解叢集的效能。叢集度量會 CloudWatch 在 1 分鐘的期間傳送至。有關更多信息 CloudWatch，請參閱什麼是 Amazon CloudWatch？ 在 Amazon 用 CloudWatch 戶指南。

AWS PCS將下列量度發佈到中的 AWS/PCS命名空間 CloudWatch。他們有一個單一的維度，`ClusterId`.

名稱	描述	個單位
ActualCapacity	IdleCapacity + UtilizedCapacity	計數
CapacityUtilization	UtilizedCapacity / ActualCapacity	計數

名稱	描述	個單位
DesiredCapacity	ActualCapacity + PendingCapacity	計數
IdleCapacity	執行中但未配置給工作的執行處理計數	計數
UtilizedCapacity	執行中並配置給工作的執行處理計數	計數

使用 AWS PCS Amazon 監控實例 CloudWatch

AWSPCS視需要啟動 Amazon EC2 執行個體，以符合PCS運算節點群組中定義的擴展需求。您可以在這些執行個體使用 Amazon 執行時監控這些執行個體 CloudWatch。您可以登入執行個體並使用互動式命令列工具，來檢查執行中執行個體的記錄。不過，根據預設， CloudWatch 指標資料只會在執行個體終止後保留一段有限的時間，而執行個體記錄檔通常會與支援執行個體的EBS磁碟區一起刪除。若要在終止執行個體PCS之後保留指標或記錄資料，您可以使用EC2啟動範本在執行個體上設定 CloudWatch 代理程式。本主題提供監控執行中執行個體的概觀，並提供如何設定持續性執行個體指標和記錄的範例。

監視執行中執行

尋找AWSPCS實例

若要監視由啟動的執行個體PCS，請尋找與叢集或計算節點群組關聯的執行中執行個體。然後，在指定執行個體的EC2主控台中，檢查 [狀態] 和 [警示] 和 [監視] 區段。如果為這些執行個體設定了登入存取權，您可以連線到這些執行個體，並檢查執行個體上的各種記錄檔。如需識別哪些執行個體受管理的詳細資訊PCS，請參閱尋找運算節點群組執行個體 AWS PCS。

啟用詳細指標

根據預設，每隔 5 分鐘收集執行個體測量結果。若要以一分鐘間隔收集指標，請在計算節點群組啟動範本中啟用詳細 CloudWatch 監控。如需詳細資訊，請參閱開啟詳細 CloudWatch監控。

設定持續執行個體指標和記錄

您可以在執行個體上安裝和設定 Amazon CloudWatch 代理程式，以保留執行個體的指標和日誌。這包括三個主要步驟：

1. 建立 CloudWatch 代理程式組態。
2. 將組態儲存在PCS執行處理可擷取的位置。
3. 撰寫可安裝 CloudWatch 代理程式軟體、擷取組態並使用組態EC2啟動 CloudWatch 代理程式的啟動範本。

如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的使用 CloudWatch 代理程式收集指標、日誌和追蹤使用 Amazon EC2 啟動模板 AWS PCS。

建立 CloudWatch 代理程式組態

在執行個體上部署 CloudWatch 代理程式之前，您必須產生JSON組態檔，以指定要收集的指標、記錄檔和追蹤。您可以使用精靈或使用文字編輯器手動建立組態檔案。此示範將手動建立組態檔案。

在已AWSCLI安裝的電腦上，建立名為 config.json 的 CloudWatch 組態檔案，其中包含下列內容。您也可以使用URL以下命令下載文件的副本。

```
https://aws-hpc-recipes.s3.amazonaws.com/main/recipes/pcs/cloudwatch/assets/config.json
```

備註

- 範例檔案中的日誌路徑適用於 Amazon Linux 2。如果您的執行個體將使用不同的基礎作業系統，請視需要變更路徑。
- 若要擷取其他記錄，請在下方新增其他項目`collect_list`。
- 中的值`{brackets}`是範本化變數。如需支援變數的完整清單，請參閱 Amazon CloudWatch 使用者指南中的手動建立或編輯 CloudWatch 代理程式組態檔案。
- 您可以選擇省略，`logs`或者`metrics`如果您不想收集這些資訊類型。

```
{
    "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
            }
        }
    }
}
```

此檔案會指示 CloudWatch 代理程式監視數個檔案，這些檔案有助於診斷執行個體啟動安裝、驗證和登入，以及其他疑難排解網域中的錯誤。其中包含：

- `/var/log/cloud-init.log`— 執行個體組態初始階段的輸出
- `/var/log/cloud-init-output.log`— 執行個體組態期間執行的命令輸出
- `/var/log/amazon/pcs/bootstrap.log`— 執行個體組態期間執行的PCS特定作業的輸出
- `/var/log/slurmd.log`— 從 Slurm 工作負載管理器的守護進程 slurmd 輸出
- `/var/log/messages`— 來自核心、系統服務和應用程式的系統訊息
- `/var/log/secure`— 與驗證嘗試相關的記錄，例如 SSH sudo 和其他安全事件

記錄檔會傳送至名為的 CloudWatch 記錄群組`/PCSLogs/instances`。日誌串流是執行處理 ID 和日誌檔基礎名稱的組合。記錄群組的保留時間為 30 天。

此外，檔案還會指示 CloudWatch 代理程式收集數個常用指標，並依執行個體 ID 彙總它們。

儲存組態

CloudWatch 代理程式組態檔必須儲存在可供PCS運算節點執行個體存取的位置。有兩種常見的方法可以做到這一點。您可以將其上傳到運算節點群組執行個體可透過其執行個體設定檔存取的 Amazon S3 儲存貯體，或者，也可以將其作為SSM參數存放在 Amazon Systems Manager 參數存放區中。

上傳到 S3 儲存貯體

若要將檔案存放在 S3 中，請使用下列AWSCLI命令。執行指令之前，請進行下列取代項目：

- Replace (取代) *DOC-EXAMPLE-BUCKET* 使用您自己的 S3 存儲桶名稱

首先，（如果您有現有存儲桶，這是可選的），請創建一個存儲桶來保存您的配置文件。

```
aws s3 mb s3://DOC-EXAMPLE-BUCKET
```

接下來，將文件上傳到存儲桶。

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

儲存為SSM參數

若要將檔案儲存為SSM參數，請使用下列指令。執行指令之前，請進行下列取代項目：

- Replace (取代) *region-code* 與您正在使用的AWS地區AWSPCS。
- (選擇性) 取代 *AmazonCloudWatch-PCS* 使用您自己的參數名稱。請注意，如果您變更名稱的前置詞，則需要在節點群組執行個體設定檔中特別新增對SSM參數的讀取存取權限。AmazonCloudWatch-

```
aws ssm put-parameter \
   --region region-code \
   --name "AmazonCloudWatch-PCS" \
   --type String \
   --value file://config.json
```

撰寫EC2啟動範本

啟動範本的特定詳細資料取決於您的組態檔案是儲存在 S3 還是SSM。

使用存放在 S3 中的組態

此指令碼會安裝 CloudWatch 代理程式、從 S3 儲存貯體匯入組態檔案，然後使用它啟動 CloudWatch 代理程式。以您自己的詳細資料取代此指令碼中的下列值：

- *DOC-EXAMPLE-BUCKET* — 您的帳戶可以讀取的 S3 儲存貯體名稱
- */config.json* — 相對於存放組態之 S3 儲存貯體根目錄的路徑

```
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==--
```

節點群組的IAM執行個體設定檔必須具有儲存貯體的存取權。以下是上述使用者資料指令碼中儲存貯體的範例IAM政策。

```
{
    "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/*"
            ]
        }
    ]
}
```

另請注意，執行個體必須允許傳出流量到 S3 和 CloudWatch端點。這可以使用安全組或VPC端點來完成，具體取決於您的叢集架構。

使用儲存於中的組態 SSM

此指令碼會安裝 CloudWatch 代理程式、從SSM參數匯入組態檔，然後隨之啟動 CloudWatch 代理程式。以您自己的詳細資料取代此指令碼中的下列值：

- (選擇性) 取代 *AmazonCloudWatch-PCS* 使用您自己的參數名稱。

```
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==--
```

節點群組的IAM執行個體原則必須已CloudWatchAgentServerPolicy附加。

如果您的參數名稱不是以開AmazonCloudWatch-頭，則需要在節點群組執行個體設定檔中特別新增對SSM參數的讀取存取權限。這是一個示例IAM策略，說明了前綴 *DOC-EXAMPLE-PREFIX*.

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "CustomCwSsmMParamReadOnly",
      "Effect" : "Allow",
      "Action" : [
        "ssm:GetParameter"
      ],
      "Resource" : "arn:aws:ssm:*:*:parameter/DOC-EXAMPLE-PREFIX*"
    }
  ]
}
```

另請注意，執行個體必須允許輸出流量傳送至SSM和 CloudWatch端點。這可以使用安全組或VPC端點來完成，具體取決於您的叢集架構。

記錄 AWS 平行運算服務API呼叫 AWS CloudTrail

AWS PCS與 (提供中的使用者 AWS CloudTrail、角色或服務所採取的動作記錄) 的 AWS 服務整合 AWS PCS。 CloudTrail 擷取 AWS PCS為事件的所有API呼叫。擷取的呼叫包括來自 AWS PCS主控台的呼叫和對 AWS PCSAPI作業的程式碼呼叫。如果您建立追蹤，您可以啟用持續交付 CloudTrail 事件到 Amazon S3 儲存貯體，包括 AWS PCS. 如果您未設定追蹤，您仍然可以在 [事件歷程記錄] 中檢視 CloudTrail 主控台中最近的事件。使用收集的資訊 CloudTrail，您可以判斷提出的要求 AWS PCS、提出要求的 IP 位址、提出要求的人員、提出要求的時間，以及其他詳細資訊。

若要進一步了解 CloudTrail，請參閱使AWS CloudTrail 用者指南。

AWS PCS中的資訊 CloudTrail

CloudTrail 在您創建帳戶 AWS 帳戶 時啟用。當活動發生在中時 AWS PCS，該活動會與事件歷史記錄中的其他 AWS 服務 CloudTrail 事件一起記錄在事件中。您可以查看，搜索和下載最近的事件 AWS 帳戶。如需詳細資訊，請參閱使用 CloudTrail 事件歷程記錄檢視事件。

對於您的事件的持續記錄 AWS 帳戶，包括事件 AWS PCS，請創建一個跟踪。追蹤可 CloudTrail 將日誌檔交付到 Amazon S3 儲存貯體。依預設，當您在主控台中建立追蹤時，該追蹤會套用至所有的 AWS 區域。追蹤記錄來自 AWS 分區中所有區域的事件，並將日誌檔傳送到您指定的 Amazon S3 儲存貯體。此外，您還可以設定其他 AWS 服務，以進一步分析 CloudTrail 記錄中收集的事件資料並採取行動。如需詳細資訊，請參閱下列內容：

- 建立追蹤的概觀
- CloudTrail 支援的服務與整合
- 設定 Amazon SNS 通知 CloudTrail
- 從多個區域接收 CloudTrail 日誌文件並從多個帳戶接收 CloudTrail 日誌文件

所有 AWS PCS動作均由記錄， CloudTrail 並記錄在AWS 平行運算服務API參考中。例如，呼叫`CreateComputeNodeGroupUpdateQueue`、和`DeleteCluster`動作會在 CloudTrail 記錄檔中產生項目。

每一筆事件或日誌專案都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 要求是使用 root 或 AWS Identity and Access Management (IAM) 使用者認證提出的。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。
- 請求是否由其他 AWS 服務提出。

如需詳細資訊，請參閱CloudTrail userIdentity元素。

瞭解 CloudTrail 記錄檔項目 AWS PCS

追蹤是一種組態，可讓事件以日誌檔的形式傳遞至您指定的 S3 儲存貯體。 CloudTrail 記錄檔包含一或多個記錄項目。事件代表來自任何來源的單一請求，包括有關請求的操作，動作的日期和時間，請求參數等信息。 CloudTrail 日誌文件不是公共API調用的有序堆棧跟踪，因此它們不會以任何特定順序顯示。

下列範例顯示CreateQueue動作的 CloudTrail 記錄項目。

```
{
    "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"
}
```

的端點和服務配額 AWS PCS

下列各節說明 AWS 平行運算服務 (AWS PCS) 的端點和服務配額。服務配額 (先前稱為限制) 是您的服務資源或作業數目上限 AWS 帳戶。

您的每個 AWS 服務都 AWS 帳戶 有預設配額。除非另有說明，否則每個配額都是區域特定規定。您可以請求提高某些配額，而其他配額無法提高。

如需詳細資訊，請參閱 AWS 一般參考中的 AWS 服務配額。

內容

服務端點

區域名稱	區域	端點	通訊協定
美國東部 (維吉尼亞北部)	us-east-1	電腦東部-1. 亞馬遜	HTTPS
美國東部 (俄亥俄)	us-east-2	電腦東部-2. 亞馬遜	HTTPS
美國西部 (奧勒岡)	us-west-2	美國西部-2. 亞馬遜	HTTPS
亞太區域 (新加坡)	ap-southeast-1	电脑公司-东南部-亚马逊	HTTPS
亞太區域 (悉尼)	ap-southeast-2	电脑公司-东南部-亚马逊	HTTPS
亞太區域 (東京)	ap-northeast-1	太平洋电脑-东北部-亚马逊	HTTPS

區域名稱	區域	端點	通訊協定
歐洲 (法蘭克福)	eu-central-1	歐盟-中央電腦 1. 亞馬遜	HTTPS
歐洲 (愛爾蘭)	eu-west-1	歐盟-西部-1. 亞馬遜	HTTPS
歐洲 (斯德哥爾摩)	eu-north-1	歐盟北部 1. 亞馬遜	HTTPS

Service Quotas

名稱	預設	可調整	Description
叢集	5	是	每個叢集的最大數目 AWS 區域。

> **Note**
>
> 預設值是由設定的初始配額 AWS。這些預設值與實際套用的配額值和可能的服務配額上限不同。如需詳細資訊，請參閱《Service Quotas 使用者指南》中的 Service Quotas 術語。

這些服務配額會列在中的AWS 平行運算服務 (PCS) 底下AWS Management Console。若要針對顯示為可調整的值要求提高配額，請參閱《Service Quotas 使用者指南》中的要求提高配額。

> **Important**
>
> 請記得檢查中的目前 AWS 區域 設定 AWS Management Console。

內部配額

下列配額為內部配額，且不可調整。

名稱	預設	可調整	Description
並行叢集建立	1	否	每個`Creating`狀態中的叢集數目上限 AWS 區域。

其他 AWS 服務的相關配額

AWS PCS使用其他 AWS 服務。這些服務的服務配額會影響您的使用 AWS PCS。

影響 Amazon EC2 服務配額 AWS PCS

- 競價型執行個體
- 執行隨需執行個
- 啟動範本
- 啟動範本版本
- Amazon EC2 API 請求

如需詳細資訊，請參閱 Amazon 彈性運算雲端使用者指南中的 Amazon EC2 服務配額。

範例版 AWS PCS本說明 AMIs

AWS PCS範例AMIs有安全性修補程式的每晚發行節奏。這些增量安全性修補程式不包含在官方發行說明中。

> **Important**
>
> 範例AMIs僅用於示範目的，不建議用於生產工作負載。

內容

AWS PCS思倫 23.11 AMI 的示例 x86_64 (Amazon Linux 2)

本文件說明 AWS PCS範例 x86_64 AMI (Amazon Linux 2) 的最新變更、新增內容、已知問題和修正。

- 創建日期：二零二四年七月十五日
- 發行日期：二○二四年八月二十二日
- 最後更新日期：二○二四年八月二十二

AMI名稱

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

支持的EC2實例

- 所有具有 64 位元 x86 處理器的執行個體。若要尋找相容的執行個體，請導覽至 Amazon EC2 主控台。選擇「執行個體類型」，然後搜尋`Architectures=x86_64`。

AMI內容

- 支持的 AWS 服務： AWS PCS
- 操作系統：Amazon 2

- 運算架構：
- Linux 核心
- EBS磁碟區類型：GP2
- AWS PCS思倫 11 安裝程式：11 月 9 日
- AWS PCS軟體安裝程式：1.0.0-1
- EFA安裝程式
- GDRCopy:
- NVIDIA驅動程式
- NVIDIACUDA: 12.2.2_535.104.05

注意

- 無

發行日期：

Updated

- 無。首次發行。

已新增

- 無。首次發行。

已移除

- 無。首次發行。

AWS PCSAMI對於泥漿 23.11 (Amazon Linux 2) 的示例

本文件說明 AWS PCS範例 Arm64 AMI (Amazon Linux 2) 的最新變更、新增內容、已知問題和修正。

- 創建日期：二零二四年七月十五日
- 發行日期：二○二四年八月二十二日

- 最後更新日期：二 ○ 二四年八月二十二

AMI名稱

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

支持的EC2實例

- 所有具有 64 位元 ARM 處理器的執行個體。若要尋找相容的執行個體，請導覽至 Amazon EC2 主控台。選擇「執行個體類型」，然後搜尋`Architectures=arm64`。

AMI內容

- 支持的 AWS 服務： AWS PCS
- 操作系統：Amazon 2
- 計算體系結構：
- Linux 核心版本:
- EBS磁碟區類型：GP2
- AWS PCS思倫 11 安裝程式：11 月 9 日
- AWS PCS軟體安裝程式：1.0.0-1
- EFA安裝程式
- GDRCopy:
- NVIDIA驅動程式
- NVIDIACUDA: 12.2.2_535.104.05

注意

- 無

發行日期：

Updated

- 無。首次發行。

已新增

- 無。首次發行。

已移除

- 無。首次發行。

AWS PCS 使用者指南的文件歷程記錄

下表說明的文件版本 AWS PCS。

日期	變更	說明文件更新	API版本已更新
2024年8月28日	已新增受管理策略頁	如需詳細資訊，請參閱AWSAWS 平行運算服務的管理原則。	N/A
2024年8月28日	AWS PCS釋放	AWS PCS使用者指南的初始版本。	AWS SDK:

AWS 詞彙表

有關最新 AWS 術語，請參閱AWS 詞彙表 參考文獻中的AWS 詞彙表。

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。