Guia do usuário

AWS PCS

AWS PCS: Guia do usuário

Table of Contents

O que é serviço de computação AWS paralela?

AWS O Serviço de Computação Paralela (AWS PCS) é um serviço gerenciado que facilita a execução e a escalabilidade de cargas de trabalho de computação de alto desempenho (HPC) e a criação de modelos científicos e de engenharia AWS usando o Slurm. Use o AWS PCS para criar clusters de computação que integram a melhor AWS computação, armazenamento, rede e visualização da categoria. Execute simulações ou crie modelos científicos e de engenharia. Simplifique e simplifique suas operações de cluster usando recursos integrados de gerenciamento e observabilidade. Capacite seus usuários a se concentrarem em pesquisa e inovação, permitindo que eles executem seus aplicativos e trabalhos em um ambiente familiar.

Tópicos

Conceitos em AWS PCS

Um cluster no AWS PCS tem 1 ou mais filas, associadas a pelo menos 1 grupo de nós de computação. Os trabalhos são enviados para filas e executados em EC2 instâncias definidas por grupos de nós de computação. Você pode usar essas bases para implementar arquiteturas de HPC sofisticadas.

Cluster

Um cluster é um recurso para gerenciar recursos e executar cargas de trabalho. Um cluster é um recurso AWS PCS que define um conjunto de configurações de computação, rede, armazenamento, identidade e agendador de tarefas. Você cria um cluster especificando qual agendador de trabalhos deseja usar (Slurm atualmente), qual configuração de agendador deseja, qual controlador de serviço deseja gerenciar o cluster e em qual VPC você deseja que os recursos do cluster sejam lançados. O agendador aceita e agenda trabalhos e também inicia os nós de computação (EC2 instâncias) que processam esses trabalhos.

Grupo de nós de computação

Um grupo de nós de computação é uma coleção de nós de computação que o AWS PCS usa para executar trabalhos ou fornecer acesso interativo a um cluster. Ao definir um grupo de nós de computação, você especifica características comuns, como tipos de EC2 instância da Amazon, contagem mínima e máxima de instâncias, sub-redes VPC de destino, Amazon Machine Image

(AMI), opção de compra e configuração de lançamento personalizada. AWS O PCS usa essas configurações para iniciar, gerenciar e encerrar com eficiência os nós de computação em um grupo de nós de computação.

Fila

Quando quiser executar um trabalho em um cluster específico, você o envia para uma fila específica (também chamada de partição). O trabalho permanece na fila até que o AWS PCS o programe para execução em um grupo de nós de computação. Você associa um ou mais grupos de nós de computação a cada fila. É necessária uma fila para agendar e executar trabalhos nos recursos do grupo de nós de computação subjacentes usando várias políticas de agendamento oferecidas pelo agendador de trabalhos. Os usuários não enviam trabalhos diretamente para um nó de computação ou grupo de nós de computação.

Administrador de sistema

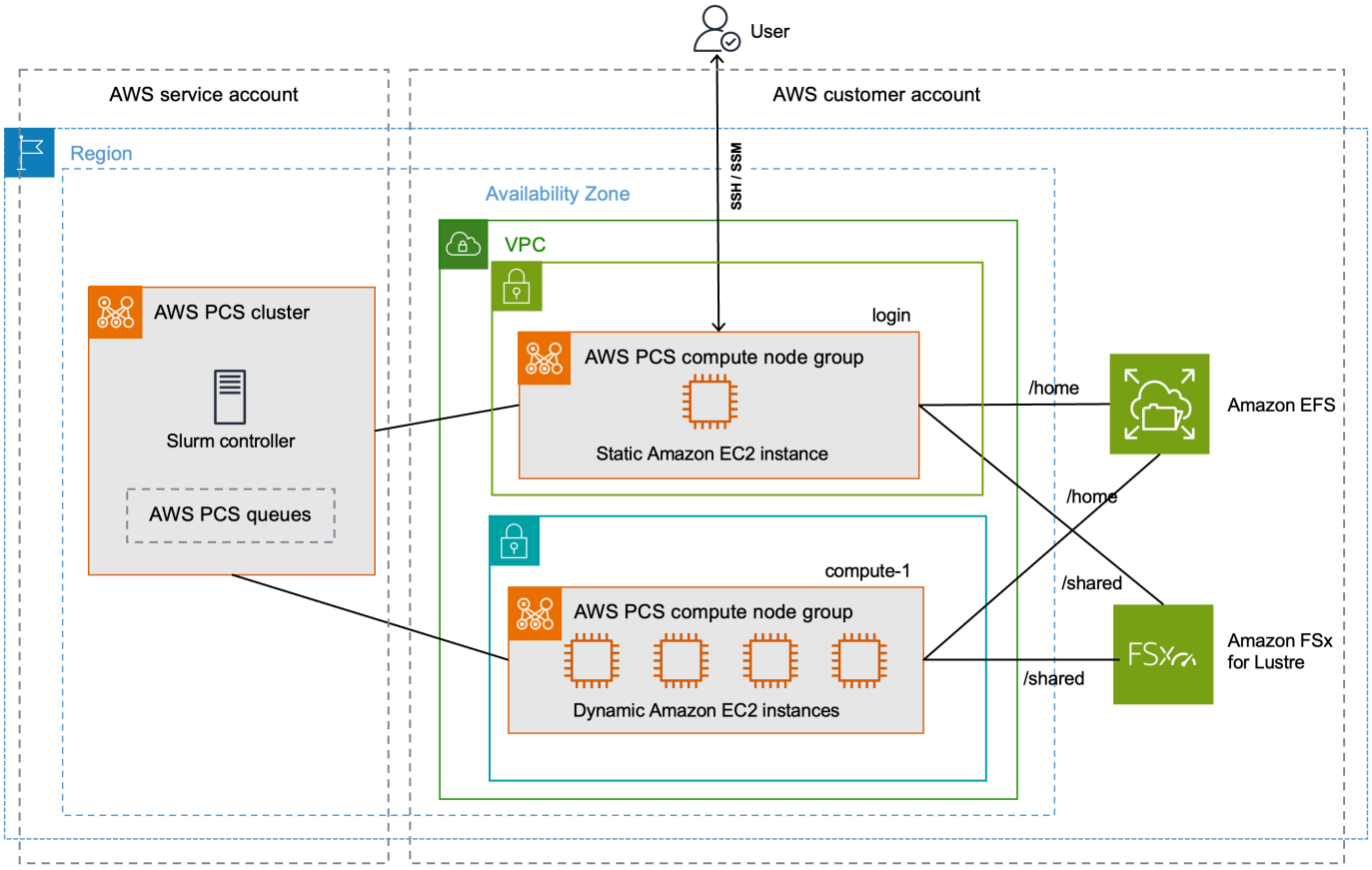
Um administrador do sistema implanta, mantém e opera um cluster. Eles podem acessar o AWS PCS por meio da AWS Management Console API AWS PCS e do AWS SDK. Eles têm acesso a clusters específicos por meio de SSH ou AWS Systems Manager, onde podem executar tarefas administrativas, executar trabalhos, gerenciar dados e realizar outras atividades baseadas em shell. Para obter mais informações, consulte a Documentação do AWS Systems Manager.

Usuário final

Um usuário final não tem a day-to-day responsabilidade de implantar ou operar um cluster. Eles usam uma interface de terminal (como SSH) para acessar recursos do cluster, executar trabalhos, gerenciar dados e realizar outras atividades baseadas em shell.

Comece a usar o serviço de computação AWS paralela

Este é um tutorial para criar um cluster simples que você pode usar para testar o AWS PCS. A figura a seguir mostra o design do cluster.



O tutorial de design de cluster tem os seguintes componentes principais:

- Uma VPC e sub-redes que atendem aos requisitos AWS de rede do PCS.
- Um sistema de arquivos Amazon EFS, que será usado como um diretório inicial compartilhado.
- Um sistema de arquivos Amazon FSx for Lustre, que fornece um diretório compartilhado de alto desempenho.
- Um cluster AWS PCS, que fornece um controlador Slurm.
- 2 grupos de nós de computação AWS PCS.
  - O grupo de `login` nós, que fornece acesso interativo baseado em shell ao sistema.
  - O grupo de `compute-1` nós fornece instâncias com escalabilidade elástica para executar trabalhos.

- 1 fila que envia trabalhos para EC2 instâncias no grupo de `compute-1` nós.

O cluster exige AWS recursos adicionais, como grupos de segurança, funções do IAM e modelos de EC2 execução, que não são mostrados no diagrama.

> **Note**
>
> Recomendamos que você conclua as etapas da linha de comando neste tópico em um shell do Bash. Se não estiver utilizando um shell Bash, alguns comandos de script, como caracteres de continuação de linha e a forma como as variáveis são definidas e utilizadas, exigirão o ajuste do seu shell. Além disso, as regras de citação e de escape do seu shell podem ser diferentes. Para obter mais informações, consulte Aspas e literais com cadeias de caracteres no Guia do AWS CLIAWS Command Line Interface usuário da versão 2.

Tópicos

Pré-requisitos para começar a usar o PCS AWS

Consulte os tópicos a seguir para preparar seu ambiente Conta da AWS de desenvolvimento local para o AWS PCS.

Tópicos

Inscreva-se AWS e crie um usuário administrativo

Conclua as tarefas a seguir para configurar o Serviço de Computação AWS Paralela (AWS PCS).

Tópicos

Inscreva-se para um Conta da AWS

Se você não tiver um Conta da AWS, conclua as etapas a seguir para criar um.

Para se inscrever em um Conta da AWS

1. Abra a https://portal.aws.amazon.com/billing/inscrição.
2. Siga as instruções online.

   Parte do procedimento de inscrição envolve receber uma chamada telefônica ou uma mensagem de texto e inserir um código de verificação pelo teclado do telefone.

   Quando você se inscreve em um Conta da AWS, um Usuário raiz da conta da AWSé criado. O usuário-raiz tem acesso a todos os Serviços da AWS e recursos na conta. Como prática recomendada de segurança, atribua o acesso administrativo a um usuário e use somente o usuário-raiz para executar tarefas que exigem acesso de usuário-raiz.

AWS envia um e-mail de confirmação após a conclusão do processo de inscrição. A qualquer momento, você pode visualizar a atividade atual da sua conta e gerenciar sua conta acessando https://aws.amazon.com/e escolhendo Minha conta.

Criar um usuário com acesso administrativo

Depois de se inscrever em um Conta da AWS, proteja seu Usuário raiz da conta da AWS AWS IAM Identity Center, habilite e crie um usuário administrativo para que você não use o usuário root nas tarefas diárias.

Proteja seu Usuário raiz da conta da AWS

1. Faça login AWS Management Consolecomo proprietário da conta escolhendo Usuário raiz e inserindo seu endereço de Conta da AWS e-mail. Na próxima página, insira a senha.

   Para obter ajuda ao fazer login usando o usuário-raiz, consulte Fazer login como usuário-raiz no Guia do usuário do Início de Sessão da AWS .
2. Habilite a autenticação multifator (MFA) para o usuário-raiz.

   Para obter instruções, consulte Habilitar um dispositivo de MFA virtual para seu usuário Conta da AWS raiz (console) no Guia do usuário do IAM.

Criar um usuário com acesso administrativo

1. Habilita o Centro de Identidade do IAM.

   Para obter instruções, consulte Habilitar o AWS IAM Identity Center no Guia do usuário do AWS IAM Identity Center .
2. No Centro de Identidade do IAM, conceda o acesso administrativo a um usuário.

   Para ver um tutorial sobre como usar o Diretório do Centro de Identidade do IAM como fonte de identidade, consulte Configurar o acesso do usuário com o padrão Diretório do Centro de Identidade do IAM no Guia AWS IAM Identity Center do usuário.

Iniciar sessão como o usuário com acesso administrativo

- Para fazer login com o seu usuário do Centro de Identidade do IAM, use o URL de login enviado ao seu endereço de e-mail quando o usuário do Centro de Identidade do IAM foi criado.

Para obter ajuda para fazer login usando um usuário do IAM Identity Center, consulte Como fazer login no portal de AWS acesso no Guia Início de Sessão da AWS do usuário.

Atribuir acesso a usuários adicionais

1. No Centro de Identidade do IAM, crie um conjunto de permissões que siga as práticas recomendadas de aplicação de permissões com privilégio mínimo.

   Para obter instruções, consulte Criar um conjunto de permissões no Guia do usuário do AWS IAM Identity Center .
2. Atribua usuários a um grupo e, em seguida, atribua o acesso de autenticação única ao grupo.

   Para obter instruções, consulte Adicionar grupos no Guia do usuário do AWS IAM Identity Center .

Instale o AWS CLI para AWS PCS

Você deve usar a versão mais recente do AWS CLI. Para obter informações, consulte Instalar ou atualizar para a versão mais recente do AWS CLI no Guia AWS Command Line Interface do Usuário da Versão 2.

Você deve configurar AWS CLI o. Para obter mais informações, consulte Configurar o AWS CLI no Guia AWS Command Line Interface do usuário para a versão 2.

Digite o seguinte comando em um prompt de comando para verificar seu AWS CLI; ele deve exibir informações de ajuda.

```
aws pcs help
```

Permissões do IAM necessárias para AWS PCS

O diretor de segurança do IAM que você está usando deve ter permissões para trabalhar com funções do IAM do AWS PCS, funções vinculadas ao serviço AWS CloudFormation, uma VPC e recursos relacionados. Para obter mais informaçõesIdentity and Access Management for AWS Parallel Computing Service, consulte e Criar uma função vinculada ao serviço no Guia do AWS Identity and Access Management usuário. Você deve concluir todas as etapas deste manual como o mesmo usuário. Execute o seguinte comando para verificar o usuário atual:

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

Usando AWS CloudFormation com o tutorial AWS PCS

O tutorial do AWS PCS tem várias etapas e tem como objetivo ajudá-lo a entender as partes de um cluster AWS PCS e os procedimentos necessários para criá-lo. Recomendamos que você siga as etapas do tutorial pelo menos uma vez. Depois de ter uma boa compreensão do que está envolvido, você pode usar AWS CloudFormation para criar rapidamente o cluster de amostra com automação.

AWS CloudFormation é um AWS serviço que permite criar e provisionar implantações de AWS infraestrutura de forma previsível e repetida. Você pode usar um CloudFormation modelo para provisionar automaticamente os AWS recursos para o cluster de amostra como uma única unidade, chamada de pilha. Você pode excluir a pilha quando terminar de usá-la.

Para obter mais informações, consulte Comece a usar um AWS CloudFormationAWS PCS.

Crie uma VPC e sub-redes para PCS AWS

Você pode criar uma VPC e sub-redes com um modelo. CloudFormation Use o URL a seguir para baixar o CloudFormation modelo e, em seguida, faça o upload do modelo no AWS CloudFormation console para criar uma nova CloudFormation pilha. Para obter mais informações, consulte Usando o AWS CloudFormation console no Guia AWS CloudFormation do usuário.

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

Com o modelo aberto no AWS CloudFormation console, insira as seguintes opções. Você pode usar os valores padrão fornecidos no modelo.

- Em Forneça um nome de pilha:
  - Em Nome da pilha, digite:

    ```
    hpc-networking
    ```
- Em Parâmetros:
  - Em VPC:

    - Em CidrBlock, insira:

      ```
      10.3.0.0/16
      ```

  - Em Sub-redes A:
    - Em CidrPublicSubnetA, insira:

      ```
      10.3.0.0/20
      ```

    - Em CidrPrivateSubnetA, insira:

      ```
      10.3.128.0/20
      ```

  - Em Sub-redes B:
    - Em CidrPublicSubnetB, insira:

      ```
      10.3.16.0/20
      ```

    - Em CidrPrivateSubnetB, insira:

      ```
      10.3.144.0/20
      ```

  - Em Sub-redes C:
    - Para ProvisionSubnetsC, selecione True
    - Em CidrPublicSubnetC, insira:

      ```
      10.3.32.0/20
      ```

    - Em CidrPrivateSubnetC, insira:

      ```
      10.3.160.0/20
      ```

- Em Capacidades:
  - Marque a caixa “Eu reconheço que isso AWS CloudFormation pode criar recursos do IAM”.

Monitore o status da CloudFormation pilha. Quando chegar`CREATE_COMPLETE`, encontre o ID do grupo de segurança padrão na nova VPC. Você usa o ID posteriormente no tutorial.

Encontre o grupo de segurança padrão para o cluster VPC

Para encontrar o ID do grupo de segurança padrão na nova VPC, siga este procedimento:

- Navegue até o console da Amazon VPC.
- No painel da VPC, selecione Filtrar por VPC.
  - Escolha a VPC com a qual o nome começa. `hpc-networking`
  - Em Segurança, escolha Grupos de segurança.
- Encontre o ID do grupo de segurança para o grupo chamado`default`. Tem a descrição`default VPC security group`. Você usa o ID posteriormente para configurar modelos de EC2 lançamento.

Crie grupos de segurança para AWS PCS

AWS O PCS depende de grupos de segurança para gerenciar o tráfego de rede que entra e sai de um cluster e seus grupos de nós de computação. Para obter informações detalhadas sobre esse tópico, consulteRequisitos e considerações do grupo de segurança.

Nesta etapa, você usará um CloudFormation modelo para criar dois grupos de segurança.

- Um grupo de segurança de cluster, que permite a comunicação entre o controlador AWS PCS, os nós de computação e os nós de login.
- Um grupo de segurança SSH de entrada, que você pode adicionar opcionalmente aos seus nós de login para oferecer suporte ao acesso SSH

Crie os grupos de segurança para AWS PCS

Você pode usar um CloudFormation modelo para criar os grupos de segurança. Use o URL a seguir para baixar o CloudFormation modelo e, em seguida, faça o upload do modelo no AWS CloudFormation console para criar uma nova CloudFormation pilha. Para obter mais informações, consulte Usando o AWS CloudFormation console no Guia AWS CloudFormation do usuário.

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

Com o modelo aberto no AWS CloudFormation console, insira as seguintes opções. Observe que algumas opções serão pré-preenchidas no modelo — você pode simplesmente deixá-las como valores padrão.

- Em Forneça um nome de pilha
  - Em Nome da pilha, digite:

    ```
    getstarted-sg
    ```

- Em Parâmetros
  - Em VpcId, escolha a VPC com a qual o nome começa. `hpc-networking`
  - (Opcional) Em ClientIpCidr, insira um intervalo de IP mais restritivo para o grupo de segurança SSH de entrada. Recomendamos que você restrinja isso com seu próprio IP/sub-rede (x.x.x.x/32 para seu próprio ip ou x.x.x.x/24 para intervalo. Substitua x.x.x.x pelo seu próprio IP PÚBLICO. Você pode obter seu IP público usando ferramentas como https://ifconfig.co/)

Monitore o status da CloudFormation pilha. Quando chega ao grupo `CREATE_COMPLETE` de segurança, os recursos estão prontos.

Há dois grupos de segurança criados, com os nomes:

- `cluster-getstarted-sg`— este é o grupo de segurança do cluster
- `inbound-ssh-getstarted-sg`— este é um grupo de segurança para permitir acesso SSH de entrada

Crie um cluster no AWS PCS

No AWS PCS, um cluster é um recurso persistente para gerenciar recursos e executar cargas de trabalho. Você cria um cluster para um agendador específico (o AWS PCS atualmente oferece suporte ao Slurm) em uma sub-rede de uma VPC nova ou existente. O cluster aceita e agenda trabalhos e também inicia os nós de computação (EC2 instâncias) que processam esses trabalhos.

Para criar um cluster

1. Abra o console AWS PCS e escolha Criar cluster.
2. Na seção Detalhes do cluster, insira os seguintes campos:

- Nome do cluster — Enter `get-started`
- Scheduler — Selecione a versão 24.11 do Slurm
- Tamanho do controlador — Selecione Pequeno

3. Na seção Rede, selecione valores para os seguintes campos:
   - VPC — Escolha a VPC chamada `hpc-networking:Large-Scale-HPC`
   - Sub-rede — Selecione a sub-rede em que o nome começa com `hpc-networking:PrivateSubnetA`
   - Grupos de segurança — Selecione o grupo de segurança do cluster chamado `cluster-getstarted-sg`
4. Selecione Criar cluster.

> **Note**
>
> O campo Status mostra Como criar enquanto o cluster está sendo provisionado. A criação do cluster pode levar vários minutos.

Crie armazenamento compartilhado para AWS PCS no Amazon Elastic File System

O Amazon Elastic File System (Amazon EFS) é um AWS serviço que fornece armazenamento de arquivos totalmente elástico e sem servidor para que você possa compartilhar dados de arquivos sem provisionar ou gerenciar a capacidade e o desempenho do armazenamento. Para obter mais informações, consulte What is Amazon Elastic File System? no Guia do usuário do Amazon Elastic File System.

O cluster de demonstração do AWS PCS usa um sistema de arquivos EFS para fornecer um diretório inicial compartilhado entre os nós do cluster. Crie um sistema de arquivos EFS na mesma VPC do seu cluster.

Como criar seu sistema de arquivos do Amazon EFS

1. Acesse o console do Amazon EFS.

2. Certifique-se de que esteja configurado da mesma forma Região da AWS em que você experimentará o AWS PCS.
3. Escolha Create file system (Criar sistema de arquivos).
4. Na página Criar sistema de arquivos, defina os seguintes parâmetros:
   - Em Nome, insira `getstarted-efs`.
   - Em Virtual Private Cloud (VPC), escolha a VPC chamada `hpc-networking:Large-Scale-HPC`
   - Escolha Criar. Isso o levará de volta à página Sistemas de arquivos.
5. Anote a ID do sistema de arquivos do sistema de `getstarted-efs` arquivos. Você usa essas informações posteriormente.

Crie armazenamento compartilhado para AWS PCS no Amazon FSx for Lustre

O Amazon FSx for Lustre torna fácil e econômico lançar e executar o popular sistema de arquivos Lustre de alto desempenho. É possível usar o Lustre para workloads em que a velocidade é importante, como machine learning, computação de alta performance (HPC), processamento de vídeo e modelagem financeira. Para obter mais informações, consulte O que é o Amazon FSx for Lustre? no Guia do usuário do Amazon FSx for Lustre.

O cluster de demonstração do AWS PCS pode usar um FSx sistema de arquivos for Lustre para fornecer um diretório compartilhado de alto desempenho entre os nós do cluster. Crie um sistema de arquivos FSx for Lustre na mesma VPC do seu cluster.

Para criar seu sistema de arquivos FSx for Lustre

1. Acesse o FSx console da Amazon.
2. Verifique se o console está configurado para usar o Região da AWS mesmo que seu cluster.
3. Escolha Create file system (Criar sistema de arquivos).
   - Em Selecionar tipo de sistema de arquivos, escolha Amazon FSx for Lustre e, em seguida, escolha Avançar.
4. Na página Especificar detalhes do sistema de arquivos, defina os seguintes parâmetros:
   - Em Detalhes do sistema de arquivos

- Em Nome, insira `getstarted-fsx`.
- Para o tipo de implantação e armazenamento, escolha Persistente, SSD
- Para taxa de transferência por unidade de armazenamento, escolha 125 MB/s/TiB
- Em Capacidade de armazenamento, insira 1,2 TiB
- Para Configuração de metadados, escolha Automático
- Para Tipo de compactação de dados, escolha LZ4

- Em Rede e segurança
  - Para Virtual Private Cloud (VPC), escolha a VPC chamada `hpc-networking:Large-Scale-HPC`
  - Para grupos de segurança da VPC, deixe o grupo de segurança chamado `default`
  - Em Sub-rede, escolha a sub-rede em que o nome começa com `hpc-networking:PrivateSubnetA`
- Deixe as outras opções definidas com seus valores padrão.
- Escolha Próximo.

5. Na página Revisar e criar, escolha Criar sistema de arquivos. Isso o levará de volta à página Sistemas de arquivos.
6. Navegue até a página de detalhes do sistema de arquivos FSx for Lustre que você criou.
7. Anote a ID do sistema de arquivos e o nome da montagem. Você usa essas informações posteriormente.

> **Note**
>
> O campo Status mostra Criando enquanto o sistema de arquivos está sendo provisionado. A criação do sistema de arquivos pode levar vários minutos. Espere até que ele seja concluído antes de continuar com o restante do tutorial.

Crie grupos de nós de computação no AWS PCS

Um grupo de nós de computação é uma coleção virtual de nós de computação (EC2 instâncias) que o AWS PCS inicia e gerencia. Ao definir um grupo de nós de computação, você especifica características comuns, como tipos de EC2 instância, contagem mínima e máxima de instâncias, sub-redes VPC de destino, opção de compra preferencial e configuração de execução

personalizada. AWS O PCS inicia, gerencia e encerra com eficiência os nós de computação em um grupo de nós de computação, de acordo com essas configurações. O cluster de demonstração usa um grupo de nós de computação para fornecer nós de login para acesso do usuário e um grupo de nós de computação separado para processar trabalhos. Os tópicos a seguir descrevem os procedimentos para configurar esses grupos de nós de computação em seu cluster.

Tópicos

Crie um perfil de instância para AWS PCS

Os grupos de nós de computação exigem um perfil de instância quando são criados. Se você usar o AWS Management Console para criar uma função para a Amazon EC2, o console cria automaticamente um perfil de instância e dá a ele o mesmo nome da função. Para obter mais informações, consulte Como usar perfis de instância no Guia AWS Identity and Access Management do usuário.

No procedimento a seguir, você usa o AWS Management Console para criar uma função para a Amazon EC2, que também cria o perfil de instância para seus grupos de nós de computação.

Para criar a função e o perfil da instância

- Navegue até o console do IAM.
- Em Access management (Gerenciamento de acesso), escolha Policies (Políticas).
  - Escolha Create policy (Criar política).
  - Em Especificar permissões, em Editor de políticas, escolha JSON.
  - Substitua o conteúdo do editor de texto pelo seguinte:

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

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

- Escolha Próximo.
- Em Revisar e criar, em Nome da política, insira`AWSPCS-getstarted-policy`.
- Escolha Criar política.

- Em Access management (Gerenciamento de acesso), escolha Roles (Funções).
- Selecione Criar perfil.
- Em Selecionar entidade confiável:
  - Para Tipo de entidade confiável, selecione AWS serviço
  - Em Caso de uso, selecione EC2.
    - Em seguida, em Escolha um caso de uso para o serviço especificado, escolha EC2.
  - Escolha Próximo.
- Em Adicionar permissões:
  - Em Políticas de permissões, pesquise por AWSPCS-getstarted-policy.
  - Marque a caixa ao lado AWSPCS-getstartedde -policy para adicioná-la à função.
  - Em Políticas de permissões, pesquise por Amazon SSMManaged InstanceCore.
  - Marque a caixa ao lado da Amazon SSMManaged InstanceCore para adicioná-la à função.
  - Escolha Próximo.
- Em Nome, revise e crie:
  - Em Detalhes da função:
    - Em Nome do perfil, insira `AWSPCS-getstarted-role`.
  - Selecione Criar perfil.

Crie modelos de lançamento para AWS PCS

Ao criar um grupo de nós de computação, você fornece um modelo de EC2 execução que o AWS PCS usa para configurar as EC2 instâncias que ele executa. Isso inclui configurações como grupos de segurança e scripts que são executados quando a instância é executada.

Nesta etapa, um CloudFormation modelo será usado para criar dois modelos de EC2 lançamento. Um modelo será usado para criar nós de login e o outro será usado para criar nós de computação. A principal diferença entre eles é que os nós de login podem ser configurados para permitir acesso SSH de entrada.

Acesse o CloudFormation modelo

Use o URL a seguir para baixar o CloudFormation modelo e, em seguida, faça o upload do modelo no AWS CloudFormation console para criar uma nova CloudFormation pilha. Para obter mais informações, consulte Usando o AWS CloudFormation console no Guia AWS CloudFormation do usuário.

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

Use o CloudFormation modelo para criar modelos de EC2 lançamento

Use o procedimento a seguir para preencher o CloudFormation modelo no AWS CloudFormation console

- Em Forneça um nome de pilha:
  - Em Nome da pilha, insira`getstarted-lt`.
- Em Parâmetros:
  - Em Segurança
    - Para VpcSecurityGroupId, selecione o grupo de segurança nomeado `default` em seu cluster VPC.
    - Para ClusterSecurityGroupId, selecione o grupo chamado `cluster-getstarted-sg`
    - Para SshSecurityGroupId, selecione o grupo chamado `inbound-ssh-getstarted-sg`
    - Para SshKeyName, selecione seu par de chaves SSH preferido.
  - Em Sistemas de arquivos
    - Para EfsFilesystemId, insira a ID do sistema de arquivos EFS que você criou anteriormente no tutorial.
    - Para FSxLustreFilesystemId, insira o ID do sistema de arquivos do FSx Lustre que você criou anteriormente no tutorial.
    - Para FSxLustreFilesystemMountName, insira o nome de montagem para o mesmo FSx sistema de arquivos Lustre.

- Escolha Avançar e, em seguida, escolha Avançar novamente.
- Selecione Enviar.

Monitore o status da CloudFormation pilha. Quando chega, `CREATE_COMPLETE` o modelo de lançamento está pronto para ser usado.

> **Note**
>
> Para ver todos os recursos criados pelo CloudFormation modelo, abra o AWS CloudFormation console. Escolha a pilha `getstarted-lt` e depois a guia Resources (Recursos).

Crie um grupo de nós de computação para nós de login no AWS PCS

Um grupo de nós de computação é uma coleção virtual de nós de computação (EC2 instâncias) que o AWS PCS inicia e gerencia. Ao definir um grupo de nós de computação, você especifica características comuns, como tipos de EC2 instância, contagem mínima e máxima de instâncias, sub-redes VPC de destino, opção de compra preferencial e configuração de execução personalizada. AWS O PCS inicia, gerencia e encerra com eficiência os nós de computação em um grupo de nós de computação, de acordo com essas configurações.

Nesta etapa, você iniciará um grupo de nós de computação estático que fornece acesso interativo ao cluster. Você pode usar o SSH ou o Amazon EC2 Systems Manager (SSM) para fazer login nele, depois executar comandos de shell e gerenciar trabalhos do Slurm.

Para criar o grupo de nós de computação

- Abra o console AWS PCS e navegue até Clusters.
- Selecione o cluster chamado `get-started`
- Navegue até grupos de nós de computação e escolha Criar.
- Na seção Configuração do grupo de nós de computação, forneça o seguinte:
  - Nome do grupo de nós de computação — Enter`login`.
- Em Configuração de computação, insira ou selecione estes valores:
  - EC2 modelo de lançamento — Escolha o modelo de lançamento em que o nome está `login-getstarted-lt`

  - Perfil da instância do IAM — Escolha o perfil da instância chamado `AWSPCS-getstarted-role`
  - Sub-redes — Selecione a sub-rede com a qual o nome começa. `hpc-networking:PublicSubnetA`
  - Instâncias — Selecione`c6i.xlarge`.
  - Configuração de escalabilidade — Em Contagem mínima de instâncias, insira. 1 Em Contagem máxima de instâncias, insira**1**.
- Em Configurações adicionais, especifique o seguinte:
  - ID da AMI — Selecione uma AMI que você deseja usar, que tenha um nome no seguinte formato:

    ```
    aws-pcs-sample_ami-amzn2-platform-slurm-version
    ```

    Para obter mais informações sobre a amostra AMIs, consulteUsando amostras de Amazon Machine Images (AMIs) com AWS PCS.
- Escolha Criar grupo de nós de computação.

O campo Status mostra Criando enquanto o grupo de nós de computação está sendo provisionado. Você pode prosseguir para a próxima etapa do tutorial enquanto ele estiver em andamento.

Crie um grupo de nós de computação para executar trabalhos de computação no PCS AWS

Nesta etapa, você iniciará um grupo de nós de computação que se expande elasticamente para executar trabalhos enviados ao cluster.

Para criar o grupo de nós de computação

- Abra o console AWS PCS e navegue até Clusters.
- Selecione o cluster chamado `get-started`
- Navegue até grupos de nós de computação e escolha Criar.
- Na seção Configuração do grupo de nós de computação, forneça o seguinte:
  - Nome do grupo de nós de computação — Enter`compute-1`.
- Em Configuração de computação, insira ou selecione estes valores:

  - EC2 modelo de lançamento — Escolha o modelo de lançamento em que o nome está `compute-getstarted-lt`
  - Perfil da instância do IAM — Escolha o perfil da instância chamado `AWSPCS-getstarted-role`
  - Sub-redes — Selecione a sub-rede com a qual o nome começa. `hpc-networking:PrivateSubnetA`
  - Instâncias — Selecione`c6i.xlarge`.
  - Configuração de escalabilidade — Em Contagem mínima de instâncias, insira. `0` Em Contagem máxima de instâncias, insira**4**.
- Em Configurações adicionais, especifique o seguinte:
  - ID da AMI — Selecione uma AMI que você deseja usar, que tenha um nome no seguinte formato:

    ```
    aws-pcs-sample_ami-amzn2-platform-slurm-version
    ```

    Para obter mais informações sobre a amostra AMIs, consulteUsando amostras de Amazon Machine Images (AMIs) com AWS PCS.
- Escolha Criar grupo de nós de computação.

O campo Status mostra Criando enquanto o grupo de nós de computação está sendo provisionado.

> **Important**
>
> Aguarde até que o campo Status mostre Ativo antes de prosseguir para a próxima etapa deste tutorial.

Crie uma fila para gerenciar trabalhos no AWS PCS

Você envia um trabalho para uma fila para executá-lo. O trabalho permanece na fila até que o AWS PCS o programe para execução em um grupo de nós de computação. Cada fila está associada a um ou mais grupos de nós de computação, que fornecem as EC2 instâncias necessárias para fazer o processamento.

Nesta etapa, você criará uma fila que usa o grupo de nós de computação para processar trabalhos.

Para criar uma fila

- Abra o console AWS PCS.
- Selecione o cluster chamado `get-started`.
- Navegue até grupos de nós de computação e verifique se o status do `compute-1` grupo é Ativo.

> **Important**
>
> O status do `compute-1` grupo deve ser Ativo antes de você prosseguir para a próxima etapa.

- Navegue até Filas e escolha Criar fila.
  - Na seção Configuração da fila, forneça os seguintes valores:
    - Nome da fila — Insira o seguinte: `demo`
    - Grupos de nós de computação — Selecione o grupo de nós de computação chamado. `compute-1`
- Selecione Criar fila.

O campo Status mostra Criando enquanto a fila está sendo criada.

> **Important**
>
> Aguarde até que o campo Status mostre Ativo antes de prosseguir para a próxima etapa deste tutorial.

Conecte-se ao seu cluster AWS PCS

Depois que o status do grupo de nós de `login` computação se tornar Ativo, você poderá se conectar à EC2 instância que ele criou.

Para se conectar ao nó de login

- Abra o console AWS PCS e navegue até Clusters.
- Selecione o cluster chamado `get-started`.
- Escolha grupos de nós de computação.

- Navegue até o grupo de nós de computação chamado`login`.
- Encontre o ID do grupo de nós de computação.
- Em outra janela ou guia do navegador, abra o EC2 console da Amazon.
  - Selecione Instances (Instâncias).
  - Pesquise EC2 instâncias com a seguinte tag. *`node-group-id`*Substitua pelo valor do ID do grupo de nós de computação da etapa anterior. Deve haver 1 instância.

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

  - Conecte-se à EC2 instância. Você pode usar o Gerenciador de Sessões ou o SSH.

    Session Manager

    - Selecione a instância.
    - Selecione Conectar.
    - Em Conectar à instância, selecione Gerenciador de sessões.
    - Selecione Conectar.
    - Selecione Conectar. Um terminal interativo é iniciado em seu navegador.

    SSH

    - Selecione a instância.
    - Selecione Conectar.
    - Em Connect to instance, selecione Cliente SSH.
    - Siga as instruções fornecidas pelo console.

      > **Note**
      >
      > O nome de usuário da instância **`ec2-user`**não é`root`.

Explore o ambiente de cluster no AWS PCS

Depois de fazer login no cluster, você pode executar comandos shell. Por exemplo, você pode alterar usuários, trabalhar com dados em sistemas de arquivos compartilhados e interagir com o Slurm.

Alterar usuário

Se você fez login no cluster usando o Gerenciador de Sessões, você pode estar conectado como `ssm-user`. Esse é um usuário especial criado para o Gerenciador de Sessões. Mude para o usuário padrão no Amazon Linux 2 usando o comando a seguir. Você não precisará fazer isso se estiver conectado usando SSH.

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

Trabalhe com sistemas de arquivos compartilhados

Você pode confirmar se o sistema de arquivos EFS e FSx os sistemas de arquivos Lustre estão disponíveis com o comando. `df -h` A saída em seu cluster deve ser semelhante à seguinte:

```
[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
```

O `/home` sistema de arquivos monta 127.0.0.1 e tem uma capacidade muito grande. Esse é o sistema de arquivos EFS que você criou anteriormente no tutorial. Todos os arquivos gravados aqui estarão disponíveis `/home` em todos os nós do cluster.

O `/shared` sistema de arquivos monta um IP privado e tem uma capacidade de 1,2 TB. Esse é o sistema FSx de arquivos do Lustre que você criou anteriormente no tutorial. Todos os arquivos gravados aqui estarão disponíveis `/shared` em todos os nós do cluster.

Interaja com o Slurm

Tópicos

- Listar filas e nós

- Mostrar empregos

Listar filas e nós

Você pode listar as filas e os nós aos quais elas estão associadas ao usosinfo. A saída do seu cluster deve ser semelhante à seguinte:

```
[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 ~]$
```

Observe a partição chamadademo. Seu status é up e tem no máximo 4 nós. Está associado aos nós do grupo de compute-1 nós. Se você editar o grupo de nós de computação e aumentar o número máximo de instâncias para 8, o número de nós será lido 8 e a lista de nós será lidacompute-1-[1-8]. Se você criasse um segundo grupo de nós de computação chamado test com 4 nós e o adicionasse à demo fila, esses nós também apareceriam na lista de nós.

Mostrar empregos

Você pode listar todos os trabalhos, em qualquer estado, no sistema comsqueue. A saída do seu cluster deve ser semelhante à seguinte:

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

Tente executar squeue novamente mais tarde, quando você tiver um trabalho do Slurm pendente ou em execução.

Execute uma tarefa de nó único no AWS PCS

Para executar um trabalho usando o Slurm, você prepara um script de envio especificando os requisitos do trabalho e o envia para uma fila com o comando. sbatch Normalmente, isso é feito em um diretório compartilhado para que os nós de login e computação tenham um espaço comum para acessar arquivos.

Conecte-se ao nó de login do seu cluster e execute os comandos a seguir em seu prompt de shell.

- Torne-se o usuário padrão. Mude para o diretório compartilhado.

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

- Use os comandos a seguir para criar um exemplo de script de trabalho:

```
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
```

- Envie o script do trabalho para o agendador do Slurm:

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

- Quando o trabalho for enviado, ele retornará uma ID do trabalho como um número. Use esse ID para verificar o status do trabalho. *job-id*Substitua o comando a seguir pelo número retornado desbatch.

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

  Example

```
squeue --job 1
```

  O squeue comando retorna uma saída semelhante à seguinte:

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

- Continue verificando o status da tarefa até que ela atinja o status R (em execução). O trabalho é feito quando squeue não devolve nada.
- Inspecione o conteúdo do /shared diretório.

```
ls -alth /shared
```

A saída do comando é semelhante à seguinte:

```
-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
```

Os arquivos `single.1.err` foram nomeados `single.1.out` e gravados por um dos nós de computação do seu cluster. Como o trabalho foi executado em um diretório compartilhado (`/shared`), eles também estão disponíveis em seu nó de login. É por isso que você configurou um sistema de arquivos FSx for Lustre para esse cluster.

- Inspecione o conteúdo do `single.1.out` arquivo.

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

A saída é semelhante à seguinte:

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

Execute uma tarefa MPI de vários nós com o Slurm no PCS AWS

Essas instruções demonstram o uso do Slurm para executar uma tarefa de interface de passagem de mensagens (MPI) no PCS. AWS

Execute os comandos a seguir em um prompt de shell do seu nó de login.

- Torne-se o usuário padrão. Mude para seu diretório inicial.

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

- Crie o código-fonte na linguagem de programação C.

```
cat > hello.c << EOF
// * mpi-hello-world - https://www.mpitutorial.com
```

```
// Released under MIT License
//
// Copyright (c) 2014 MPI Tutorial.
//
// Permission is hereby granted, free of charge, to any person obtaining a copy
// of this software and associated documentation files (the "Software"), to
// deal in the Software without restriction, including without limitation the
// rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
// sell copies of the Software, and to permit persons to whom the Software is
// furnished to do so, subject to the following conditions:
// The above copyright notice and this permission notice shall be included in
// all copies or substantial portions of the Software.
//
// THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
// IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
// FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
// AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
// LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
// FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
// DEALINGS IN THE SOFTWARE.

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

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

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

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

  // Get the name of the processor
  char processor_name[MPI_MAX_PROCESSOR_NAME];
  int name_len;
  MPI_Get_processor_name(processor_name, &name_len);
```

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

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

- Carregue o módulo openMPI.

```
module load openmpi
```

- Compile o programa C.

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

- Escreva um script de envio de trabalhos no Slurm.

```
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
```

- Mude para o diretório compartilhado.

```
cd /shared
```

- Envie o roteiro do trabalho.

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

- Use `squeue` para monitorar o trabalho até que seja concluído.
- Confira o conteúdo de`multi.out`:

```
cat multi.out
```

A saída é semelhante à seguinte. Observe que cada classificação tem seu próprio endereço IP porque foi executada em um nó diferente.

```
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
```

Exclua seus AWS recursos para AWS PCS

Depois de concluir os grupos de clusters e nós que você criou para este tutorial, você deve excluir os recursos que você criou.

> **Important**
>
> Você recebe cobranças de cobrança por todos os recursos em execução no seu Conta da AWS

Para excluir recursos do AWS PCS que você criou para este tutorial

- Abra o console AWS PCS.
- Navegue até o cluster chamado get-started.
- Navegue até a seção Filas.
- Selecione a fila chamada demo.
- Escolha Excluir.

> **Important**
>
> Espere até que a fila seja excluída antes de continuar.

- Navegue até a seção Grupos de nós de computação.
- Selecione o grupo de nós de computação chamado compute-1.

- Escolha Excluir.
- Selecione o grupo de nós de computação chamado login.
- Escolha Excluir.

> **Important**
>
> Espere até que os dois grupos de nós de computação tenham sido excluídos antes de continuar.

- Na página de detalhes do cluster para começar, escolha Excluir.

> **Important**
>
> Espere até que o cluster seja excluído antes de prosseguir com as etapas subsequentes.

Para excluir outros AWS recursos que você criou para este tutorial

- Abra o console do IAM.
  - Escolha Perfis.
  - Selecione a função chamada AWSPCS-getstarted-role e escolha Excluir.
  - Depois que a função for excluída, escolha Políticas.
  - Selecione a política chamada AWSPCS-getstarted-policy e escolha Excluir.
- Abra o console de AWS CloudFormation.
  - Selecione a pilha chamada getstarted-lt.
  - Escolha Excluir.

> **Important**
>
> Aguarde até que a pilha seja excluída antes de continuar.

- Abra o Console do Amazon EFS .
  - Escolha Sistemas de arquivos.
  - Selecione o sistema de arquivos chamado getstarted-efs.
  - Escolha Excluir.

> **Important**
>
> Aguarde até que o sistema de arquivos seja excluído antes de continuar.

- Abra o FSx console da Amazon.
  - Escolha Sistemas de arquivos.
  - Selecione o sistema de arquivos chamado getstarted-fsx.
  - Escolha Excluir.

> **Important**
>
> Aguarde até que o sistema de arquivos seja excluído antes de continuar.

- Abra o console de AWS CloudFormation.
  - Selecione a pilha chamada getstarted-sg.
  - Escolha Excluir.
- Abra o console de AWS CloudFormation.
  - Selecione a pilha chamada hpc-networking.
  - Escolha Excluir.

Comece a usar um AWS CloudFormationAWS PCS

Você pode usar AWS CloudFormation para criar um cluster AWS PCS. AWS CloudFormation permite criar e provisionar implantações de AWS infraestrutura de forma previsível e repetida. Você pode usar AWS CloudFormation para provisionar automaticamente recursos de vários AWS serviços para criar aplicativos altamente confiáveis, escaláveis e econômicos Nuvem AWS sem criar e configurar a infraestrutura subjacente. AWS AWS CloudFormation permite que você use um arquivo de modelo para criar e excluir uma coleção de recursos juntos como uma única unidade, chamada de pilha. Para obter mais informações sobre AWS CloudFormation, consulte O que é AWS CloudFormation? no Guia do AWS CloudFormation usuário. Para obter mais informações sobre os tipos de recursos AWS PCS em AWS CloudFormation, consulte a referência do tipo de recurso AWS PCS no Guia AWS CloudFormation do usuário.

Tópicos

Use AWS CloudFormation para criar um cluster AWS PCS de amostra

O procedimento a seguir usa um CloudFormation modelo no AWS Management Console para criar um cluster AWS PCS de amostra. Para obter mais informações sobre AWS CloudFormation, consulte O que é AWS CloudFormation? no Guia do AWS CloudFormation usuário. Para obter mais informações sobre os tipos de recursos AWS PCS em AWS CloudFormation, consulte a referência do tipo de recurso AWS PCS no Guia AWS CloudFormation do usuário.

Para criar o cluster de amostra

1. Escolha o Região da AWS para criar o cluster (o link abre o CloudFormation console com o modelo):

    - Leste dos EUA (Norte da Virgínia) (us-east-1)

- Leste dos EUA (Ohio) (us-east-2)
- Oeste dos EUA (Oregon) (us-west-2)
- Ásia-Pacífico (Cingapura) (ap-southeast-1)
- Ásia-Pacífico (Sydney) (ap-southeast-2)
- Ásia-Pacífico (Tóquio) (ap-northeast-1)
- Europa (Frankfurt) (eu-central-1)
- Europa (Irlanda) (eu-west-1)
- Europa (Londres) (eu-west-2)
- Europa (Estocolmo) (eu-north-1)
- AWS GovCloud (Leste dos EUA) (us-gov-east-1)
- AWS GovCloud (Oeste dos EUA) (us-gov-west-1)

2. Em Forneça um nome de pilha, insira um nome descritivo. Esse é o nome da sua CloudFormation pilha. O modelo usa esse valor como o nome do seu cluster AWS PCS.
3. Em Parâmetros:

   a. Em SlurmVersion, escolha a versão do Slurm que você deseja que seu cluster use.

   b. Em NodeArchitecture, escolha x86 para implantar um cluster que usa instâncias compatíveis com x86_64 ou escolha Graviton para usar instâncias Arm64.

   c. Para KeyName, escolha um par de chaves SSH para acessar os nós de login do cluster. Verifique se você tem o arquivo PEM do par de chaves escolhido.

   d. Para ClientIpCidr, insira um intervalo de IP no formato CIDR para controlar o acesso aos nós de login.

   > **Warning**
   >
   > O valor padrão de `0.0.0.0/0` permite o acesso de todos os endereços IP.

   e. Deixe os valores para o HpcRecipesS3Bucket e HpcRecipesBranchcomo seus valores padrão.

4. Em Capacidades e transformações:

   a. Marque a caixa de seleção para confirmar que AWS CloudFormation criará recursos do IAM.

b. Marque a caixa de seleção para confirmar que AWS CloudFormation criará recursos do IAM com nomes personalizados.

c. Marque a caixa de seleção `CAPABILITY_AUTO_EXPAND` para confirmar a nova pilha. Para obter mais informações, consulte CreateStack na Referência de APIs do AWS CloudFormation .

5. Selecione Criar pilha.

6. Monitore o status da sua pilha. Você pode se conectar ao cluster depois que o status da pilha for`CREATE_COMPLETE`.

Conecte-se a um cluster AWS PCS criado com AWS CloudFormation

Depois de criar um cluster AWS PCS a partir de um AWS CloudFormation modelo, você pode usar o console AWS PCS (no AWS Management Console) para administrar o cluster. Você também pode se conectar a um dos nós de login do cluster para administrar o cluster, executar trabalhos e gerenciar dados. A AWS CloudFormation pilha fornece links que você pode usar para se conectar ao seu cluster.

Para se conectar ao seu cluster

1. Abra o console do AWS CloudFormation.

2. Escolha a pilha que você criou.

3. Escolha a guia Saídas da pilha.

   A pilha fornece os seguintes links:

   - PcsConsoleUrl— Escolha este link para abrir o console AWS PCS com o cluster selecionado. Você pode usá-lo para explorar as configurações de cluster, grupo de nós e fila.
   - Ec2 ConsoleUrl — Escolha esse link para abrir o EC2 console da Amazon, filtrado para mostrar as instâncias que o grupo de nós de login do cluster gerencia.

     Nessa visualização, você pode selecionar uma instância e escolher Connect. A instância do cluster de amostra oferece suporte a SSH de entrada e AWS Systems Manager conexões em um navegador da web. Para obter mais informações, consulte Conecte-se ao seu cluster AWS PCS.

Depois de se conectar a uma instância de login, você pode seguir o tutorial emExplore o ambiente de cluster no AWS PCS.

Limpe um cluster AWS PCS em AWS CloudFormation

Se você AWS CloudFormation costumava criar seu cluster AWS PCS, você pode abrir o AWS CloudFormation console e excluir a pilha para excluir o cluster e todos os recursos associados.

> **Important**
>
> Para o cluster de amostra, se você criou grupos ou filas de nós de computação adicionais em seu cluster (além dos `compute-1` grupos `login` e criados pelo CloudFormation modelo de amostra), você deve usar o console AWS PCS ou AWS CLI excluir esses recursos antes de excluir a CloudFormation pilha. Para obter mais informações, consulte Excluindo um cluster no AWS PCS.

Partes de um CloudFormation modelo para AWS PCS

Um CloudFormation modelo tem 1 ou mais seções, cada uma com uma finalidade específica. AWS CloudFormation define formato, sintaxe e linguagem padrão em um modelo. Para obter mais informações, consulte Como trabalhar com CloudFormation modelos no Guia AWS CloudFormation do usuário.

CloudFormation os modelos são altamente personalizáveis e, portanto, seus formatos podem variar. Para entender as partes necessárias de um CloudFormation modelo para criar um cluster AWS PCS, recomendamos que você examine o modelo de amostra que fornecemos para criar um cluster de amostra. Este tópico explica resumidamente as seções desse modelo de amostra.

> **Important**
>
> Os exemplos de código neste tópico não estão completos. A presença de ellipsis (`[...]`) indica que há um código adicional que não é exibido. Para baixar o CloudFormation modelo completo em formato YAML, consulte. AWS CloudFormation modelos para criar um cluster AWS PCS de amostra

Sumário

Cabeçalho

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

`AWSTemplateFormatVersion`identifica a versão do formato do modelo com a qual o modelo está em conformidade. Para obter mais informações, consulte a sintaxe da versão do formato de CloudFormation modelo no Guia do AWS CloudFormation usuário.

`Transform`especifica uma macro que é CloudFormation usada para processar o modelo. Para obter mais informações, consulte a seção Transformação do CloudFormation modelo no Guia AWS CloudFormation do usuário. A `AWS::Serverless-2016-10-31` transformação permite AWS CloudFormation processar um modelo escrito na sintaxe AWS Serverless Application Model (AWS SAM). Para obter mais informações, consulte `AWS::Serverless`transform no Guia AWS CloudFormation do usuário.

Metadados

```
### Stack metadata
Metadata:
  AWS::CloudFormation::Interface:
    ParameterGroups:
      - Label:
          default: PCS Cluster configuration
        Parameters:
          - SlurmVersion
          - ManagedAccounting
          - AccountingPolicyEnforcement
      - Label:
```

```
          default: PCS ComputeNodeGroups configuration
        Parameters:
          - NodeArchitecture
          - KeyName
          - ClientIpCidr
      - Label:
          default: HPC Recipes configuration
        Parameters:
          - HpcRecipesS3Bucket
          - HpcRecipesBranch
```

A `metadata` seção de um CloudFormation modelo fornece informações sobre o próprio modelo. O modelo de amostra cria um cluster completo de computação de alto desempenho (HPC) que usa AWS PCS. A seção de metadados do modelo de amostra declara parâmetros que controlam como AWS CloudFormation inicia (provisiona) a pilha correspondente. Existem parâmetros que controlam a escolha da arquitetura (`NodeArchitecture`), a versão do Slurm (`SlurmVersion`) e os controles de acesso (`KeyNameeClientIpCidr`).

Parâmetros

A `Parameters` seção define os parâmetros personalizados para o modelo. AWS CloudFormation usa essas definições de parâmetros para criar e validar o formulário com o qual você interage ao iniciar uma pilha a partir desse modelo.

```
Parameters:

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

  SlurmVersion:
    Type: String
    Default: 24.11
    Description: Version of Slurm to use
    AllowedValues:
         - 24.05
         - 24.11
```

```
 ManagedAccounting:
   Type: String
   Default: 'disabled'
   AllowedValues:
     - 'enabled'
     - 'disabled'
   Description: Monitor cluster usage, manage access control, and enforce resource
limits with Slurm accounting. Requires Slurm 24.11 or newer.

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

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

 ClientIpCidr:
   Description: IP(s) allowed to access the login node over SSH. We recommend that
you restrict it with your own IP/subnet (x.x.x.x/32 for your own ip or x.x.x.x/24 for
range. Replace x.x.x.x with your own PUBLIC IP. You can get your public IP using tools
such as https://ifconfig.co/)
   Default: 127.0.0.1/32
   Type: String
   AllowedPattern: (\d{1,3})\.(\d{1,3})\.(\d{1,3})\.(\d{1,3})/(\d{1,2})
   ConstraintDescription: Value must be a valid IP or network range of the form
x.x.x.x/x.

 HpcRecipesS3Bucket:
   Type: String
   Default: aws-hpc-recipes
   Description: HPC Recipes for AWS S3 bucket
   AllowedValues:
        - aws-hpc-recipes
        - aws-hpc-recipes-dev
 HpcRecipesBranch:
   Type: String
   Default: main
   Description: HPC Recipes for AWS release branch
```

```
    AllowedPattern: '^(?!.*/\.git$)(?!.*/\.)(?!.*\\.\.)[a-zA-Z0-9-_\.]+$'
```

Mapeamentos

A `Mappings` seção define pares de valores-chave que especificam valores com base em determinadas condições ou dependências.

```
Mappings:

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

Recursos

A `Resources` seção declara os AWS recursos a serem provisionados e configurados como parte da pilha.

```
Resources:

  [...]
```

O modelo provisiona a infraestrutura de cluster de amostra em camadas. Tudo começa com a `Networking` configuração da VPC. O armazenamento é fornecido por sistemas duplos: `EfsStorage` para armazenamento compartilhado e `FSxLStorage` para armazenamento de alto desempenho. O cluster principal é estabelecido por meio de`PCSCluster`.

```

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

```
        ProvisionSubnetsC: "False"
      TemplateURL: !Sub 'https://${HpcRecipesS3Bucket}.s3.amazonaws.com/
${HpcRecipesBranch}/recipes/net/hpc_large_scale/assets/main.yaml'

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

  FSxLStorage:
    Type: AWS::CloudFormation::Stack
    Properties:
      Parameters:
        PerUnitStorageThroughput: 125
        SubnetId: !GetAtt [ Networking, Outputs.DefaultPrivateSubnet ]
        VpcId: !GetAtt [ Networking, Outputs.VPC ]
      TemplateURL: !Sub 'https://${HpcRecipesS3Bucket}.s3.amazonaws.com/
${HpcRecipesBranch}/recipes/storage/fsx_lustre/assets/persistent.yaml'

  [...]

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

Para recursos de computação, o modelo cria dois grupos de nós: `PCSNodeGroupLogin` para um único nó de login e `PCSNodeGroupCompute` para até quatro nós de computação. Esses grupos de

nós são suportados `PCSInstanceProfile` por permissões e, `PCSLaunchTemplate` por exemplo, configurações.

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

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

  # Compute Node groups - Login Nodes
  PCSNodeGroupLogin:
    Type: AWS::PCS::ComputeNodeGroup
    Properties:
      ClusterId: !GetAtt [PCSCluster, Id]
      Name: login
```

```
      ScalingConfiguration:
        MinInstanceCount: 1
        MaxInstanceCount: 1
      IamInstanceProfileArn: !GetAtt [ PCSInstanceProfile, Outputs.InstanceProfileArn ]
      CustomLaunchTemplate:
        TemplateId: !GetAtt [ PCSLaunchTemplate, Outputs.LoginLaunchTemplateId ]
        Version: 1
      SubnetIds:
        - !GetAtt [ Networking, Outputs.DefaultPublicSubnet ]
      AmiId: !GetAtt [PcsSampleAmi, AmiId]
      InstanceConfigs:
        - InstanceType: !FindInMap [ Architecture, LoginNodeInstances, !Ref
NodeArchitecture ]

 # Compute Node groups - Compute Nodes
 PCSNodeGroupCompute:
   Type: AWS::PCS::ComputeNodeGroup
   Properties:
     ClusterId: !GetAtt [PCSCluster, Id]
     Name: compute-1
     ScalingConfiguration:
       MinInstanceCount: 0
       MaxInstanceCount: 4
     IamInstanceProfileArn: !GetAtt [ PCSInstanceProfile, Outputs.InstanceProfileArn ]
     CustomLaunchTemplate:
       TemplateId: !GetAtt [ PCSLaunchTemplate, Outputs.ComputeLaunchTemplateId ]
       Version: 1
     SubnetIds:
       - !GetAtt [ Networking, Outputs.DefaultPrivateSubnet ]
     AmiId: !GetAtt [PcsSampleAmi, AmiId]
     InstanceConfigs:
       - InstanceType: !FindInMap [ Architecture, ComputeNodeInstances, !Ref
NodeArchitecture ]
```

O agendamento de trabalhos é feito por completo. `PCSQueueCompute`

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

```
      ComputeNodeGroupConfigurations:
        - ComputeNodeGroupId: !GetAtt [PCSNodeGroupCompute, Id]
```

A seleção da AMI acontece automaticamente por meio da função Pcs AMILookup Fn Lambda e dos recursos relacionados.

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

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

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

Saídas

O modelo gera a identificação e o gerenciamento URLs do cluster por meio de `ClusterIdPcsConsoleUrl`, e. `Ec2ConsoleUrl`

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

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

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

AWS CloudFormation modelos para criar um cluster AWS PCS de amostra

Região da AWS nome	Região da AWS	Exibir fonte	Visualizar em AWS Infrastructure Composer	Pilha de lançamento
Leste dos EUA (Norte da Virgínia)	us-east-1	Baixar YAML	Visualizar em AWS Infrastructure Composer	
Leste dos EUA (Ohio)	us-east-2	Baixar YAML	Visualizar em AWS Infrastructure Composer	

Região da AWS nome	Região da AWS	Exibir fonte	Visualizar em AWS Infrastructure Composer	Pilha de lançamento
Oeste dos EUA (Oregon)	us-west-2	Baixar YAML	Visualizar em AWS Infrastructure Composer	Launch Stack
Ásia-Pacífico (Singapura)	ap-southeast-1	Baixar YAML	Visualizar em AWS Infrastructure Composer	Launch Stack
Ásia-Pacífico (Sydney)	ap-southeast-2	Baixar YAML	Visualizar em AWS Infrastructure Composer	Launch Stack
Ásia-Pacífico (Tóquio)	ap-northeast-1	Baixar YAML	Visualizar em AWS Infrastructure Composer	Launch Stack
Europa (Frankfurt)	eu-central-1	Baixar YAML	Visualizar em AWS Infrastructure Composer	Launch Stack
Europa (Irlanda)	eu-west-1	Baixar YAML	Visualizar em AWS Infrastructure Composer	Launch Stack
Europa (Londres)	eu-west-2	Baixar YAML	Visualizar em AWS Infrastructure Composer	Launch Stack
Europa (Estocolmo)	eu-north-1	Baixar YAML	Visualizar em AWS Infrastructure Composer	Launch Stack
AWS GovCloud (Leste dos EUA)	us-gov-east-1	Baixar YAML	Não compatível	Launch Stack

Região da AWS nome	Região da AWS	Exibir fonte	Visualizar em AWS Infrastructure Composer	Pilha de lançamento
AWS GovCloud (Oeste dos EUA)	us-gov-west-1	Baixar YAML	Não compatível	

AWS Clusters PCS

Um cluster AWS PCS consiste nos seguintes componentes:

- Instâncias gerenciadas do software programador do sistema HPC, como o daemon de controle Slurm (). `slurmctld`
- Componentes que se integram ao programador do sistema HPC para provisionar e gerenciar instâncias da Amazon EC2.
- Componentes que se integram ao programador do sistema HPC para transmitir registros e métricas para a Amazon. CloudWatch

Esses componentes são executados em uma conta gerenciada por AWS. Eles trabalham juntos para gerenciar as EC2 instâncias da Amazon em sua conta de cliente. AWS O PCS provisiona interfaces de rede elásticas em sua sub-rede Amazon VPC para fornecer conectividade do software agendador às EC2 instâncias da Amazon (por exemplo, para oferecer suporte ao agendamento de trabalhos em lote nelas e permitir que os usuários executem comandos do agendador para listar e gerenciar esses trabalhos).

Tópicos

Criando um cluster no AWS Parallel Computing Service

Este tópico fornece uma visão geral das opções disponíveis e descreve o que considerar ao criar um cluster no Serviço de Computação AWS Paralela (AWS PCS). Se esta é a primeira vez que você cria um cluster AWS PCS, recomendamos que você sigaComece a usar o serviço de computação AWS paralela. O tutorial pode ajudá-lo a criar um sistema de HPC funcional sem expandir para todas as opções e arquiteturas de sistema disponíveis que são possíveis.

Pré-requisitos

- Uma VPC e uma sub-rede existentes que atendem aos requisitos. AWS Rede PCS Antes de implantar um cluster para uso em ambientes de produção, convém ter uma compreensão integral dos requisitos da VPC e da sub-rede. Para criar uma VPC e uma sub-rede, consulte. Criação de uma VPC para seu AWS cluster PCS
- Um diretor do IAM com permissões para criar e gerenciar recursos do AWS PCS. Para obter mais informações, consulte Identity and Access Management for AWS Parallel Computing Service.

Crie um cluster AWS PCS

Você pode usar o AWS Management Console ou AWS CLI para criar um cluster.

AWS Management Console

Para criar um cluster

1. Abra o console AWS PCS em https://console.aws.amazon.com/pcs/home#/clusters e escolha Create cluster.
2. Na seção Configuração do cluster, insira os seguintes campos:
   - Nome do cluster — Um nome para seu cluster. O nome só pode conter caracteres alfanuméricos (sensíveis a maiúsculas e minúsculas) e hifens. Ele deve começar com um caractere alfabético e não pode ter mais de 40 caracteres. O nome deve ser exclusivo no Região da AWS e no Conta da AWS qual você está criando o cluster.
   - Agendador — Escolha um agendador e uma versão. Para obter mais informações, consulte Versões Slurm no PCS AWS.
   - Tamanho do controle — Escolha um tamanho para o controle. Isso determina quantos trabalhos e nós de computação simultâneos podem ser gerenciados pelo cluster AWS PCS. Você só pode definir o tamanho do controlador quando o cluster é criado. Para obter mais informações sobre dimensionamento, consulteTamanho do cluster no AWS PCS.
3. Na seção Rede, selecione valores para os seguintes campos:
   - VPC — Escolha uma VPC existente que atenda aos requisitos da PCS. AWS Para obter mais informações, consulte AWS Requisitos e considerações sobre PCS, VPC e sub-rede. Depois de criar o cluster, você não pode alterar sua VPC. Se nenhum VPCs estiver listado, você deverá criar um primeiro.

- Sub-rede — Todas as sub-redes disponíveis na VPC selecionada são listadas. Escolha uma sub-rede que atenda aos requisitos de sub-rede do AWS PCS. Para obter mais informações, consulte AWS Requisitos e considerações sobre PCS, VPC e sub-rede. Recomendamos que você selecione uma sub-rede privada para evitar a exposição dos endpoints do agendador à Internet pública.
- Grupos de segurança — especifique os grupos de segurança que você deseja que o AWS PCS associe às interfaces de rede que ele cria para seu cluster. Você deve selecionar pelo menos um grupo de segurança que permita a comunicação entre seu cluster e seus nós de computação. Você pode selecionar Criar rapidamente um grupo de segurança para que o AWS PCS crie um com a configuração necessária na VPC selecionada ou selecione um grupo de segurança existente. Para obter mais informações, consulte Requisitos e considerações do grupo de segurança.

4. (Opcional) Na seção Configuração da contabilidade do Slurm, você pode ativar a contabilidade do Slurm e definir os parâmetros contábeis. Para obter mais informações, consulte Contabilidade de slurm no PCS AWS.
5. (Opcional) Na seção Configuração do Slurm, você pode especificar as opções de configuração do Slurm que substituem os padrões definidos pelo PCS: AWS

   - Reduza o tempo de inatividade — isso controla por quanto tempo os nós de computação provisionados dinamicamente permanecem ativos após a conclusão ou o término dos trabalhos colocados neles. Definir isso para um valor maior pode aumentar a probabilidade de uma tarefa subsequente ser executada no nó, mas pode levar ao aumento dos custos. Um valor menor diminuirá os custos, mas poderá aumentar a proporção de tempo que seu sistema de HPC gasta provisionando nós em vez de executar trabalhos neles.
   - Prolog — Esse é um caminho totalmente qualificado para um diretório de scripts de prolog em suas instâncias do grupo de nós de computação. Isso corresponde à configuração Prolog no Slurm. Observe que isso deve ser um diretório, não um caminho para um executável específico.
   - Epilog — Esse é um caminho totalmente qualificado para um diretório de scripts de epilog em suas instâncias do grupo de nós de computação. Isso corresponde à configuração do Epilog no Slurm. Observe que isso deve ser um diretório, não um caminho para um executável específico.
   - Selecionar parâmetros de tipo — Isso ajuda a controlar o algoritmo de seleção de recursos usado pelo Slurm. Definir esse valor como `CR_CPU_Memory` ativará o agendamento com reconhecimento de memória, enquanto configurá-lo como `CR_CPU` ativará o agendamento

somente da CPU. Esse parâmetro corresponde à SelectTypeParametersconfiguração no Slurm, onde `SelectType` é definido `select/cons_tres` pelo AWS PCS.

6. (Opcional) Em Tags, adicione qualquer tag ao seu cluster AWS PCS.
7. Selecione Criar cluster. O campo Status é exibido `Creating` enquanto o AWS PCS cria o cluster. Esse processo pode levar alguns minutos.

> **Important**
>
> Só pode haver 1 cluster em um `Creating` estado Região da AWS por pessoa Conta da AWS. AWS O PCS retornará um erro se já houver um cluster em um `Creating` estado quando você tentar criar um cluster.

AWS CLI

Para criar um cluster

1. Crie o cluster usando o comando a seguir. Antes da execução do comando, realize as seguintes substituições:
   - *`region`*Substitua pelo ID do Região da AWS qual você deseja criar seu cluster, como`us-east-1`.
   - Substitua *`my-cluster`* por um nome de cluster. O nome só pode conter caracteres alfanuméricos (sensíveis a maiúsculas e minúsculas) e hifens. Ele deve começar com um caractere alfabético e não pode ter mais de 40 caracteres. O nome deve ser exclusivo dentro Região da AWS e Conta da AWS onde você está criando o cluster.
   - *`24.11`*Substitua por qualquer versão compatível do Slurm.

     > **Note**
     >
     > AWS Atualmente, o PCS suporta Slurm 24.11 e 24.05.

   - *`SMALL`*Substitua por qualquer tamanho de cluster compatível. Isso determina quantos trabalhos e nós de computação simultâneos podem ser gerenciados pelo cluster AWS PCS. Ele só pode ser definido quando o cluster é criado. Para obter mais informações sobre dimensionamento, consulteTamanho do cluster no AWS PCS.

- Substitua o valor `subnetIds` por pelo seu. Recomendamos que você selecione uma sub-rede privada para evitar a exposição dos endpoints do agendador à Internet pública.
- Especifique o `securityGroupIds` que você deseja que o AWS PCS associe às interfaces de rede que ele cria para seu cluster. Os grupos de segurança devem estar na mesma VPC do cluster. Você deve selecionar pelo menos um grupo de segurança que permita a comunicação entre seu cluster e seus nós de computação. Para obter mais informações, consulte Requisitos e considerações do grupo de segurança.
- Opcionalmente, você pode fornecer uma chave KMS personalizada para criptografar os dados do seu controlador usando. `--kms-key-id` *kms-key* *kms-key*Substitua por um ARN, ID de chave ou alias do KMS existente. Observe que a conta usada para criar o cluster deve ter `kms:Decrypt` privilégios na chave KMS personalizada.

```
aws pcs create-cluster --region region \
    --cluster-name my-cluster \
    --scheduler type=SLURM,version=24.11 \
    --size SMALL \
    --networking subnetIds=subnet-ExampleId1,securityGroupIds=sg-ExampleId1
```

- Opcionalmente, você pode adicionar a `--slurm-configration` opção de personalizar o comportamento do Slurm e especificar as opções de configuração do Slurm. O exemplo a seguir define o tempo de inatividade de redução para 60 minutos (3600 segundos), ativa a contabilização do Slurm e especifica `slurm.conf` as configurações como o valor de. `slurmCustomSettings` Para obter mais informações, consulte Contabilidade de slurm no PCS AWS.

> **Note**
>
> A contabilidade é compatível com o Slurm 24.11 ou posterior.

```
aws pcs create-cluster --region region \
    --cluster-name my-cluster \
    --scheduler type=SLURM,version=24.11 \
    --size SMALL \
    --networking subnetIds=subnet-ExampleId1,securityGroupIds=sg-ExampleId1
```

```
    --slurm-configuration
scaleDownIdleTimeInSeconds=3600,accounting='{mode=STANDARD}',slurmCustomSettings='[{p
```

2. O provisionamento do cluster pode levar vários minutos. Você pode consultar o status do cluster com o comando a seguir. Não continue criando filas ou grupos de nós de computação até que o campo de status do cluster seja exibido. `ACTIVE`

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

> **Important**
>
> Só pode haver 1 cluster em um `Creating` estado Região da AWS por pessoa Conta da AWS. AWS O PCS retornará um erro se já houver um cluster em um `Creating` estado quando você tentar criar um cluster.

Próximas etapas recomendadas para seu cluster

- Adicione grupos de nós de computação.
- Adicione filas.
- Ativar o registro em log.

Excluindo um cluster no AWS PCS

Este tópico fornece uma visão geral de como excluir um cluster do AWS PCS.

Considerações ao excluir um AWS cluster PCS

- Todas as filas associadas ao cluster devem ser excluídas antes que o cluster possa ser excluído. Para obter mais informações, consulte Excluindo uma fila no PCS AWS.
- Todos os grupos de nós de computação associados ao cluster devem ser excluídos antes que o cluster possa ser excluído. Para obter mais informações, consulte Excluindo um grupo de nós de computação no PCS AWS.

Excluir o cluster

Você pode usar o AWS Management Console ou AWS CLI para excluir um cluster.

AWS Management Console

Para excluir um cluster

1. Abra o console AWS PCS.
2. Selecione o cluster a ser excluído.
3. Escolha Excluir.
4. O campo Status do cluster é exibido`Deleting`. Pode demorar vários minutos para isso ser concluído.

AWS CLI

Para excluir um cluster

1. Use o comando a seguir para excluir um cluster, com essas substituições:
   - *region-code*Substitua por aquele em que Região da AWS seu cluster está.
   - *my-cluster*Substitua pelo nome ou ID do seu cluster.

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

2. A exclusão do cluster pode levar alguns minutos. Você pode verificar o status do seu cluster com o comando a seguir.

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

Tamanho do cluster no AWS PCS

AWS O PCS fornece clusters altamente disponíveis e seguros, ao mesmo tempo em que automatiza tarefas importantes, como aplicação de patches, provisionamento de nós e atualizações.

Ao criar um cluster, você seleciona um tamanho para ele com base em dois fatores:

- O número de nós de computação que ele gerenciará
- O número de trabalhos ativos e em fila que você espera executar no cluster

> **Important**
>
> Você não pode alterar o tamanho do cluster depois de criar o cluster. Se você precisar alterar o tamanho, deverá criar um novo cluster.

Tamanho do cluster do Slurm	Número de instâncias gerenciadas	Número de trabalhos ativos e em fila
Pequeno	Até 32	Até 256
Médio	Até 512	Até 8192
Grande	Até 2048	Até 16384

Exemplos

- Se seu cluster tiver até 24 instâncias gerenciadas e executar até 100 trabalhos, escolha Pequeno.
- Se seu cluster tiver até 24 instâncias gerenciadas e executar até 1.000 trabalhos, escolha Médio.
- Se seu cluster tiver até 1.000 instâncias gerenciadas e executar até 100 trabalhos, escolha Grande.
- Se seu cluster tiver até 1.000 instâncias gerenciadas e executar até 10.000 trabalhos, escolha Grande.

Trabalhando com segredos de cluster no AWS PCS

Como parte da criação de um cluster, o AWS PCS cria um segredo de cluster que é necessário para se conectar ao agendador de tarefas no cluster. Você também cria grupos de nós de computação AWS PCS, que definem conjuntos de instâncias a serem executadas em resposta a eventos de escalabilidade. AWS O PCS configura instâncias iniciadas por esses grupos de nós de computação com o segredo do cluster para que eles possam se conectar ao agendador de tarefas. Há casos em que talvez você queira configurar os clientes do Slurm manualmente. Os exemplos incluem a

criação de um nó de login persistente ou a configuração de um gerenciador de fluxo de trabalho com recursos de gerenciamento de tarefas.

AWS O PCS armazena o segredo do cluster como um segredo gerenciado com o prefixo `pcs!` in AWS Secrets Manager. O custo do segredo está incluído na cobrança pelo uso do AWS PCS.

> **Warning**
>
> Não modifique o segredo do seu cluster. AWS O PCS não conseguirá se comunicar com o cluster se você modificar o segredo do cluster. AWS O PCS não suporta a rotação do segredo do cluster. Você deve criar um novo cluster se precisar modificar o segredo do cluster.

Sumário

Use AWS Secrets Manager para encontrar o segredo do cluster

AWS Management Console

1. Navegue até o console do Secrets Manager.
2. Escolha Segredos e, em seguida, pesquise o `pcs!` prefixo.

> **Note**
>
> Um segredo de cluster AWS PCS tem um nome no formato em `pcs!slurm-secret-`*`cluster-id`* que *`cluster-id`* é o ID do cluster AWS PCS.

AWS CLI

Cada segredo do cluster AWS PCS também é marcado com`aws:pcs:`*`cluster-id`*. Você pode obter o ID secreto de um cluster com o comando a seguir. Faça essas substituições antes de executar o comando:

- *region*Substitua pelo Região da AWS para criar seu cluster, como`us-east-1`.
- *cluster-id*Substitua pelo ID do cluster AWS PCS para encontrar o segredo do cluster.

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

Use o AWS PCS para encontrar o segredo do cluster

Você pode usar o AWS CLI para encontrar o ARN de um segredo de cluster AWS PCS. Digite o comando a seguir, fazendo as seguintes substituições:

- *region*Substitua pelo Região da AWS para criar seu cluster, como`us-east-1`.
- *my-cluster*Substitua pelo nome ou identificador do seu cluster.

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

O exemplo de saída a seguir é do `get-cluster` comando. Vocês podem usar `secretArn` e `secretVersion` juntos para descobrir o segredo.

```
{
    "cluster": {
        "name": "get-started",
        "id": "pcs_123456abcd",
        "arn": "arn:aws:pcs:us-east-1:111122223333:cluster/pcs_123456abcd",
        "status": "ACTIVE",
        "createdAt": "2024-12-17T21:03:52+00:00",
        "modifiedAt": "2024-12-17T21:03:52+00:00",
        "scheduler": {
            "type": "SLURM",
            "version": "24.05"
        },
        "size": "SMALL",
        "slurmConfiguration": {
            "authKey": {
                "secretArn": "arn:aws:secretsmanager:us-east-1:111122223333:secret:pcs!
slurm-secret-pcs_123456abcd-a12ABC",
```

```
                "secretVersion": "ef232370-d3e7-434c-9a87-ec35c1987f75"
            }
        },
        "networking": {
            "subnetIds": [
                "subnet-0123456789abcdef0"
            ],
            "securityGroupIds": [
                "sg-0123456789abcdef0"
            ]
        },
        "endpoints": [
            {
                "type": "SLURMCTLD",
                "privateIpAddress": "10.3.149.220",
                "port": "6817"
            }
        ]
    }
}
```

Obtenha o segredo do cluster Slurm

Você pode usar o Secrets Manager para obter a versão atual codificada em base64 de um segredo de cluster do Slurm. O exemplo a seguir usa o. AWS CLI Faça as seguintes substituições antes de executar o comando.

- *region*Substitua pelo Região da AWS para criar seu cluster, como`us-east-1`.
- *secret-arn*Substitua pelo `secretArn` de um cluster AWS PCS.

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

Para obter informações sobre como usar o segredo do cluster Slurm, consulte. Usando instâncias autônomas como nós de login do AWS PCS

Permissões

Você usa um diretor do IAM para obter o segredo do cluster Slurm. O diretor do IAM deve ter permissão para ler o segredo. Para obter mais informações, consulte Termos e conceitos de funções no Guia AWS Identity and Access Management do usuário.

O exemplo de política do IAM a seguir permite o acesso a um exemplo de segredo de cluster.

```
{
    "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 Grupos de nós de computação PCS

Um grupo de nós de computação AWS PCS é uma coleção lógica de nós (EC2 instâncias da Amazon). Esses nós podem ser usados para executar trabalhos de computação, bem como para fornecer acesso interativo baseado em shell a um sistema de HPC. Um grupo de nós de computação consiste em regras para criar nós, incluindo quais tipos de EC2 instâncias da Amazon usar, quantas instâncias executar, se usar instâncias spot ou instâncias sob demanda, quais sub-redes e grupos de segurança usar e como configurar cada instância quando ela for iniciada. Quando essas regras são atualizadas, o AWS PCS atualiza os recursos associados ao grupo de nós de computação de acordo com a correspondência.

Tópicos

Criação de um grupo de nós de computação no AWS PCS

Este tópico fornece uma visão geral das opções disponíveis e descreve o que considerar ao criar um grupo de nós de computação no Serviço de Computação AWS Paralela (AWS PCS). Se esta é a primeira vez que você cria um grupo de nós de computação no AWS PCS, recomendamos que você siga o tutorial emComece a usar o serviço de computação AWS paralela. O tutorial pode ajudá-lo a criar um sistema HPC funcional sem expandir para todas as opções disponíveis e arquiteturas de sistema possíveis.

Pré-requisitos

- Cotas de serviço suficientes para iniciar o número desejado de EC2 instâncias em seu Região da AWS. Você pode usar o AWS Management Consolepara verificar e solicitar aumentos em suas cotas de serviço.
- Uma VPC e uma sub-rede existentes que atendem aos requisitos de rede do AWS PCS. Recomendamos que você entenda completamente esses requisitos antes de implantar um cluster

para uso em produção. Para obter mais informações, consulte AWS Requisitos e considerações sobre PCS, VPC e sub-rede. Você também pode usar um CloudFormation modelo para criar uma VPC e sub-redes. AWS fornece uma receita de HPC para o CloudFormation modelo. Para obter mais informações, consulte aws-hpc-recipesem GitHub.

- Um perfil de instância do IAM com permissões para chamar a ação da `RegisterComputeNodeGroupInstance` API AWS PCS e acessar quaisquer outros AWS recursos necessários para suas instâncias de grupo de nós. Para obter mais informações, consulte Perfis de instância do IAM para o AWS Parallel Computing Service.
- Um modelo de lançamento para suas instâncias de grupos de nós. Para obter mais informações, consulte Usando modelos de EC2 lançamento da Amazon com AWS PCS.
- Para criar um grupo de nós computacionais que usa instâncias Amazon EC2 Spot, você deve ter a função vinculada ao serviço AWSServiceRoleForEC2Spot em seu. Conta da AWS Para obter mais informações, consulte Função do Amazon EC2 Spot para AWS PCS.

Crie um grupo de nós de computação no AWS PCS

Você pode criar um grupo de nós de computação usando o. AWS Management Console ou o. AWS CLI

AWS Management Console

Para criar seu grupo de nós de computação usando o console

1. Abra o console AWS PCS.
2. Selecione o cluster em que você deseja criar um grupo de nós de computação. Navegue até grupos de nós de computação e escolha Criar.
3. Na seção Configuração do grupo de nós de computação, forneça um nome para seu grupo de nós. O nome só pode conter caracteres alfanuméricos e hífens que diferenciem maiúsculas e minúsculas. Ele deve começar com um caractere alfabético e não pode ter mais de 25 caracteres. O nome deve ser exclusivo dentro do cluster.
4. Em Configuração de computação, insira ou selecione estes valores:

    a. EC2 modelo de execução — Selecione um modelo de execução personalizado para usar nesse grupo de nós. Os modelos de execução podem ser usados para personalizar configurações de rede, como sub-rede e grupos de segurança, configuração de monitoramento e armazenamento em nível de instância. Se você não tiver um modelo

de lançamento preparado, consulte Usando modelos de EC2 lançamento da Amazon com AWS PCS para saber como criar um.

> **Important**
>
> AWS O PCS cria um modelo de lançamento gerenciado para cada grupo de nós de computação. Esses são nomeados`pcs-identifier-do-not-delete`. Não os selecione ao criar ou atualizar um grupo de nós de computação, ou o grupo de nós não funcionará corretamente.

b. EC2 versão do modelo de lançamento — Você deve selecionar uma versão do seu modelo de lançamento personalizado. Se você alterar a versão posteriormente, deverá atualizar o grupo de nós de computação para detectar alterações no modelo de execução. Para obter mais informações, consulte Atualização de um grupo de nós de computação AWS PCS.

c. ID de AMI — se seu modelo de lançamento não incluir um ID de AMI ou se você quiser substituir o valor no modelo de lançamento, forneça um ID de AMI aqui. Observe que a AMI usada para o grupo de nós deve ser compatível com o AWS PCS. Você também pode selecionar uma amostra de AMI fornecida por AWS. Para obter mais informações sobre esse tópico, consulteAmazon Machine Images (AMIs) para AWS PCS.

d. Perfil de instância do IAM — escolha um perfil de instância para o grupo de nós. Um perfil de instância concede à instância permissões para acessar AWS recursos e serviços com segurança. Se você não tiver um preparado, você pode selecionar Criar um perfil básico para que o AWS PCS crie um para você com a política mínima, ou consultePerfis de instância do IAM para o AWS Parallel Computing Service.

e. Sub-redes — Escolha uma ou mais sub-redes na VPC em que seu cluster PCS está implantado. AWS Se você selecionar várias sub-redes, as comunicações EFA não estarão disponíveis entre os nós, e a comunicação entre nós em sub-redes diferentes poderá aumentar a latência. Certifique-se de que as sub-redes especificadas aqui correspondam às que você define no modelo de EC2 execução.

f. Instâncias — escolha um ou mais tipos de instância para atender às solicitações de escalabilidade no grupo de nós. Todos os tipos de instância devem ter a mesma arquitetura de processador (x86_64 ou arm64) e número de v. CPUs Se as instâncias tiverem GPUs, todos os tipos de instância deverão ter o mesmo número de GPUs.

g. Configuração de escalabilidade — especifique o número mínimo e máximo de instâncias para o grupo de nós. Você pode definir uma configuração estática, na qual há um

número fixo de nós em execução, ou uma configuração dinâmica, na qual até a contagem máxima de nós pode ser executada. Para uma configuração estática, defina o mínimo e o máximo para o mesmo número, maior que zero. Para uma configuração dinâmica, defina o mínimo de instâncias como zero e o máximo de instâncias como um número maior que zero. AWS O PCS não oferece suporte a grupos de nós de computação com uma combinação de instâncias estáticas e dinâmicas.

5. (Opcional) Em Configurações adicionais, especifique o seguinte:

   a. Opção de compra — selecione entre instâncias spot e sob demanda.

   b. Estratégia de alocação — se você selecionou a opção de compra spot, pode especificar como os pools de capacidade spot são escolhidos ao iniciar instâncias no grupo de nós. Para obter mais informações, consulte Estratégias de alocação para instâncias spot no Guia do usuário do Amazon Elastic Compute Cloud. Essa opção não tem efeito se você tiver selecionado a opção de compra sob demanda.

6. (Opcional) Na seção de configurações Slurm personalizadas, forneça os seguintes valores:

   a. Peso — Esse valor define a prioridade dos nós no grupo para fins de agendamento. Os nós com pesos mais baixos têm maior prioridade e as unidades são arbitrárias. Para obter mais informações, consulte Peso na Slurm documentação.

   b. Memória real — Esse valor define o tamanho (em GB) da memória real nos nós do grupo de nós. Ele deve ser usado em conjunto com a `CR_CPU_Memory` opção na Slurm configuração de cluster no AWS PCS. Para obter mais informações, consulte a RealMemory documentação do Slurm.

7. (Opcional) Em Tags, adicione qualquer tag ao seu grupo de nós de computação.

8. Escolha Criar grupo de nós de computação. O campo Status mostra `Creating` enquanto o AWS PCS provisiona o grupo de nós. Isso pode demorar vários minutos.

Próxima etapa recomendada

- Adicione seu grupo de nós a uma fila no AWS PCS para permitir que ele processe trabalhos.

AWS CLI

Para criar seu grupo de nós de computação usando AWS CLI

Crie sua fila com o comando a seguir. Antes da execução do comando, realize as seguintes substituições:

1. *region*Substitua pelo ID do Região da AWS para criar seu cluster, como`us-east-1`.
2. *my-cluster*Substitua pelo nome ou pelo nome `clusterId` do seu cluster.
3. *my-node-group*Substitua pelo nome do seu grupo de nós de computação. O nome só pode conter caracteres alfanuméricos (sensíveis a maiúsculas e minúsculas) e hifens. Ele deve começar com um caractere alfabético e não pode ter mais de 25 caracteres. O nome deve ser exclusivo dentro do cluster.
4. *subnet-ExampleID1*Substitua por uma ou mais sub-redes IDs do seu cluster VPC.
5. *lt-ExampleID1*Substitua pelo ID do seu modelo de lançamento personalizado. Se você não tiver um preparado, veja Usando modelos de EC2 lançamento da Amazon com AWS PCS para aprender como criar um.

   > **Important**
   >
   > AWS O PCS cria um modelo de lançamento gerenciado para cada grupo de nós de computação. Esses são nomeadospcs-*identifier*`-do-not-delete`. Não os selecione ao criar ou atualizar um grupo de nós de computação, ou o grupo de nós não funcionará corretamente.

6. *launch-template-version*Substitua por uma versão específica do modelo de lançamento. AWS O PCS associa seu grupo de nós a essa versão específica do modelo de lançamento.
7. *arn:InstanceProfile*Substitua pelo ARN do seu perfil de instância do IAM. Se você não tiver um preparado, consulte Usando modelos de EC2 lançamento da Amazon com AWS PCS para obter orientação.
8. *min-instances*Substitua e *max-instances* por valores inteiros. Você pode definir uma configuração estática, na qual há um número fixo de nós em execução, ou uma configuração dinâmica, na qual até a contagem máxima de nós pode ser executada. Para uma configuração estática, defina o mínimo e o máximo para o mesmo número, maior que zero. Para uma configuração dinâmica, defina o mínimo de instâncias como zero e o máximo

de instâncias como um número maior que zero. AWS O PCS não oferece suporte a grupos de nós de computação com uma combinação de instâncias estáticas e dinâmicas.

9. *t3.large*Substitua por outro tipo de instância. Você pode adicionar mais tipos de instância especificando uma lista de `instanceType` configurações. Por exemplo, .*--instance-configs instanceType=c6i.16xlarge instanceType=c6a.16xlarge* Todos os tipos de instância devem ter a mesma arquitetura de processador (x86_64 ou arm64) e número de v. CPUs Se as instâncias tiverem GPUs, todos os tipos de instância deverão ter o mesmo número de 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
```

Há várias configurações opcionais que você pode adicionar ao `create-compute-node-group` comando.

- Você pode especificar `--amiId` se seu modelo de lançamento personalizado não inclui uma referência a uma AMI ou se você deseja substituir esse valor. Observe que a AMI usada para o grupo de nós deve ser compatível com o AWS PCS. Você também pode selecionar uma amostra de AMI fornecida por AWS. Para obter mais informações sobre esse tópico, consulteAmazon Machine Images (AMIs) para AWS PCS.
- Você pode selecionar entre instâncias sob demanda (`ONDEMAND`) e spot (`SPOT`) usando`--purchase-option`. Sob demanda é o padrão. Se você escolher instâncias spot, também poderá usar `--allocation-strategy` para definir como o AWS PCS escolhe os pools de capacidade spot ao iniciar instâncias no grupo de nós. Para obter mais informações, consulte Estratégias de alocação para instâncias spot no Guia do usuário do Amazon Elastic Compute Cloud.
- É possível fornecer opções de Slurm configuração para os nós no grupo de nós usando`--slurm-configuration`. Você pode definir o peso (prioridade de agendamento) e a memória real. Os nós com pesos mais baixos têm maior prioridade e as unidades são arbitrárias. Para obter mais informações, consulte Peso na Slurm documentação. A memória real define o tamanho (em GB) da memória real nos nós do grupo de nós. Ele deve ser usado em conjunto

com a `CR_CPU_Memory` opção do cluster no AWS PCS em sua Slurm configuração. Para obter mais informações, consulte a RealMemory documentação do Slurm.

> **Important**
>
> A criação do grupo de nós de computação pode levar vários minutos.

Você pode consultar o status do seu grupo de nós com o comando a seguir. Você não poderá associar o grupo de nós a uma fila até que seu status chegue`ACTIVE`.

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

Atualização de um grupo de nós de computação AWS PCS

Este tópico fornece uma visão geral das opções disponíveis e descreve o que considerar ao atualizar um grupo de nós computacionais do AWS PCS.

Opções para atualizar um grupo de nós computacionais do AWS PCS

A atualização de um grupo de nós computacionais do AWS PCS permite que você altere as propriedades das instâncias lançadas pelo AWS PCS, bem como as regras de como essas instâncias são lançadas. Por exemplo, você pode substituir a AMI para instâncias de grupos de nós por outra com software diferente instalado nela. Ou você pode atualizar os grupos de segurança para alterar a conectividade de rede de entrada ou saída. Você também pode alterar a configuração de escalabilidade ou até mesmo alterar a opção de compra preferencial de ou para instâncias spot.

As seguintes configurações do grupo de nós não podem ser alteradas após a criação:

- Name
- Instâncias

Considerações ao atualizar um grupo de nós de computação AWS PCS

Os grupos de nós de computação definem EC2 instâncias que são usadas para processar trabalhos, fornecer acesso interativo ao shell e outras tarefas. Eles geralmente são associados a uma ou mais filas AWS PCS. Ao atualizar seu grupo de nós de computação para alterar seu comportamento (ou o de seus nós), considere o seguinte:

- As alterações nas propriedades do grupo de nós de computação entram em vigor quando o status do grupo de nós de computação muda de Atualizando para Ativo. Novas instâncias são lançadas com as propriedades atualizadas.
- As atualizações que não afetam a configuração de nós específicos não afetam os nós em execução. Por exemplo, adicionar uma sub-rede e alterar a estratégia de alocação.
- Se você atualizar o modelo de execução de um grupo de nós de computação, deverá atualizar o grupo de nós de computação para usar a nova versão.
- Para adicionar ou remover um grupo de segurança dos nós em um grupo de nós de computação, edite seu modelo de execução e atualize o grupo de nós de computação. Novas instâncias são lançadas com o conjunto atualizado de grupos de segurança.
- Se você editar diretamente um grupo de segurança usado por um grupo de nós de computação, ele terá efeito imediato nas instâncias em execução e no futuro.
- Se você adicionar ou remover permissões do perfil de instância do IAM usado por um grupo de nós de computação, isso terá efeito imediato nas instâncias em execução e no futuro.
- Para alterar a AMI usada pelas instâncias de um grupo de nós de computação, atualize o grupo de nós de computação (ou seu modelo de execução) para usar a nova AMI e aguarde até que o AWS PCS substitua as instâncias.
- AWS O PCS substitui as instâncias existentes no grupo de nós após uma operação de atualização do grupo de nós. Se houver trabalhos em execução em um nó, esses trabalhos poderão ser concluídos antes que o AWS PCS substitua o nó. Os processos interativos do usuário (como em instâncias de nós de login) são encerrados. O status do grupo de nós retorna para `Active` quando o AWS PCS marca as instâncias para substituição, mas a substituição real ocorre quando as instâncias estão ociosas.
- Se você diminuir o número máximo de instâncias permitido em um grupo de nós de computação, o AWS PCS removerá os nós do Slurm para atingir o novo máximo. AWS O PCS encerra as instâncias em execução associadas aos nós do Slurm removidos. Os trabalhos em execução nos nós removidos falham e retornam às filas.

- AWS O PCS cria um modelo de lançamento gerenciado para cada grupo de nós de computação. Eles são nomeados`pcs-identifier-do-not-delete`. Não os selecione ao criar ou atualizar um grupo de nós de computação, ou o grupo de nós não funcionará corretamente.
- Se você atualizar um grupo de nós de computação para usar o Spot como opção de compra, deverá ter a função vinculada ao serviço AWSServiceRoleForEC2Spot em sua conta. Para obter mais informações, consulte Função do Amazon EC2 Spot para AWS PCS.

Para atualizar um grupo de nós computacionais do AWS PCS

Você pode atualizar um grupo de nós usando o AWS Management Console ou o AWS CLI.

AWS Management Console

Para atualizar um grupo de nós de computação

1. Abra o console do AWS PCS em `https://console.aws.amazon.com/pcs/home#/clusters`
2. Selecione o cluster em que você deseja atualizar um grupo de nós de computação.
3. Navegue até os grupos de nós de computação, vá até o grupo de nós que você deseja atualizar e selecione Editar.
4. Na configuração de computação, Configurações adicionais e Slurm seções de configurações de personalização, atualize todos os valores, exceto:
   - Instâncias — você não pode alterar as instâncias em um grupo de nós de computação.
5. Selecione Atualizar. O campo Status mostrará Atualizando enquanto as alterações estão sendo aplicadas.

> **Important**
>
> As atualizações do grupo de nós de computação podem levar vários minutos.

AWS CLI

Para atualizar um grupo de nós de computação

1. Atualize seu grupo de nós de computação com o comando a seguir. Antes da execução do comando, realize as seguintes substituições:

a. *region-code*Substitua pela região da AWS na qual você deseja criar seu cluster.

b. *my-node-group*Substitua pelo nome ou `computeNodeGroupId` pelo seu grupo de nós de computação.

c. *my-cluster*Substitua pelo nome ou pelo nome `clusterId` do seu cluster.

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

2. Atualize todos os parâmetros do grupo de nós, exceto `--instance-configs` o. Por exemplo, para definir um novo ID de AMI, `--amiId my-custom-ami-id` informe onde *my-custom-ami-id* é substituído pela AMI de sua escolha.

> **Important**
>
> A atualização do grupo de nós de computação pode levar vários minutos.

Você pode consultar o status do seu grupo de nós com o comando a seguir.

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

Excluindo um grupo de nós de computação no PCS AWS

Este tópico fornece uma visão geral das opções disponíveis e descreve o que considerar ao excluir um grupo de nós de computação no AWS PCS.

Considerações ao excluir um grupo de nós de computação

Os grupos de nós de computação definem EC2 instâncias que são usadas para processar trabalhos, fornecer acesso interativo ao shell e outras tarefas. Eles geralmente são associados a uma ou mais filas AWS PCS. Antes de excluir um grupo de nós de computação, considere o seguinte:

- Todas EC2 as instâncias iniciadas pelo grupo de nós de computação serão encerradas. Isso cancelará os trabalhos que estão sendo executados nessas instâncias e encerrará a execução de processos interativos.
- Você deve desassociar o grupo de nós de computação de todas as filas antes de excluí-lo. Para obter mais informações, consulte Atualizando uma fila AWS PCS.

Excluir o grupo de nós de computação

Você pode usar o AWS Management Console ou AWS CLI para excluir um grupo de nós de computação.

AWS Management Console

Para excluir um grupo de nós de computação

1. Abra o console AWS PCS.
2. Selecione o cluster do grupo de nós de computação.
3. Navegue até grupos de nós de computação e selecione o grupo de nós de computação a ser excluído.
4. Escolha Excluir.
5. O campo Status é exibido`Deleting`. Pode demorar vários minutos para isso ser concluído.

> **Note**
>
> Você pode usar comandos nativos do seu agendador para confirmar se o grupo de nós de computação foi excluído. Por exemplo, use `sinfo` ou `squeue` para Slurm.

AWS CLI

Para excluir um grupo de nós de computação

- Use o comando a seguir para excluir um grupo de nós de computação com essas substituições:
  - *`region-code`*Substitua por aquele em que Região da AWS seu cluster está.
  - *`my-node-group`*Substitua pelo nome ou ID do seu grupo de nós de computação.

- *my-cluster*Substitua pelo nome ou ID do seu cluster.

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

A exclusão do grupo de nós de computação pode levar vários minutos.

> **Note**
>
> Você pode usar comandos nativos do seu agendador para confirmar se o grupo de nós de computação foi excluído. Por exemplo, use `sinfo` ou `squeue` para Slurm.

Obtenha detalhes do grupo de nós de computação no AWS PCS

Você pode usar o AWS Management Console or AWS CLI para obter detalhes sobre um grupo de nós de computação, como o ID do grupo de nós de computação, o Amazon Resource Name (ARN) e o ID da Amazon Machine Image (AMI). Esses detalhes geralmente são valores obrigatórios para ações e configurações da API AWS PCS.

AWS Management Console

Para obter detalhes do grupo de nós de computação

1. Abra o console AWS PCS.
2. Selecione o cluster.
3. Escolha grupos de nós de computação.
4. Escolha um grupo de nós de computação no painel da lista.

AWS CLI

Para obter detalhes do grupo de nós de computação

1. Use a ação ListClustersda API para encontrar o nome ou ID do seu cluster.

```
aws pcs list-clusters
```

Exemplos de resultado:

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

2. Use a ação ListComputeNodeGroupsda API para listar os grupos de nós de computação em um cluster.

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

Exemplo de chamada:

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

Exemplos de resultado:

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

```
            "arn": "arn:aws:pcs:us-east-1:111122223333:cluster/pcs_abc1234567/
computenodegroup/pcs_abc456abc7",
            "clusterId": "pcs_abc1234567",
            "createdAt": "2025-04-01T20:19:31+00:00",
            "modifiedAt": "2025-04-01T20:19:31+00:00",
            "status": "ACTIVE"
        }
    ]
}
```

3. Use a ação GetComputeNodeGroupda API para obter detalhes adicionais de um grupo de nós de computação.

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

Exemplo de chamada:

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

Exemplos de resultado:

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

```
        "iamInstanceProfileArn": "arn:aws:iam::111122223333:instance-profile/
AWSPCS-get-started-cfn-us-east-1",
        "scalingConfiguration": {
            "minInstanceCount": 0,
            "maxInstanceCount": 4
        },
        "instanceConfigs": [
            {
                "instanceType": "c6i.xlarge"
            }
        ]
    }
}
```

Encontrando instâncias de grupos de nós de computação no AWS PCS

Cada grupo de nós de computação do AWS PCS pode iniciar EC2 instâncias com configurações compartilhadas. Você pode usar EC2 tags para encontrar instâncias em um grupo de nós de computação no AWS Management Console ou com o. AWS CLI

AWS Management Console

Para encontrar suas instâncias do grupo de nós de computação

1. Abra o console AWS PCS.
2. Selecione o cluster.
3. Escolha grupos de nós de computação.
4. Encontre o ID do grupo de nós de login que você criou.
5. Navegue até o EC2 console e escolha Instâncias.
6. Pesquise as instâncias com a seguinte tag. *node-group-id*Substitua pelo ID (não pelo nome) do seu grupo de nós de computação.

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

7. (Opcional) Você pode alterar o valor do estado da instância no campo de pesquisa para encontrar instâncias que estão sendo configuradas ou que foram encerradas recentemente.

8. Encontre o ID da instância e o endereço IP de cada instância na lista de instâncias marcadas.

AWS CLI

Para encontrar suas instâncias de grupo de nós, use os comandos a seguir. Antes de executar os comandos, faça as seguintes substituições:

- *region-code*Substitua pelo Região da AWS do seu cluster. Example: `us-east-1`
- *node-group-id*Substitua pelo ID (não pelo nome) do seu grupo de nós de computação. Para encontrar a ID de um grupo de nós de computação, consulteObtenha detalhes do grupo de nós de computação no AWS PCS.
- `running`Substitua por outros estados de instância, como `pending` ou `terminated` para encontrar EC2 instâncias em outros estados.

```
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}
```

Esse comando retorna uma saída semelhante à seguinte. O valor de `PublicIP` é `null` se a instância estiver em uma sub-rede privada.

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

> **Note**
>
> Se você espera `describe-instances` retornar um grande número de instâncias, deve usar opções para várias páginas. Para obter mais informações, consulte DescribeInstancesa Amazon Elastic Compute Cloud API Reference.

Usando modelos de EC2 lançamento da Amazon com AWS PCS

Na Amazon EC2, um modelo de lançamento pode armazenar um conjunto de preferências para que você não precise especificá-las individualmente ao iniciar instâncias. AWS O PCS incorpora modelos de lançamento como uma forma flexível de configurar grupos de nós de computação. Ao criar um grupo de nós, você fornece um modelo de lançamento. AWS O PCS cria um modelo de lançamento derivado que inclui transformações para ajudar a garantir que ele funcione com o serviço.

Entender quais são as opções e considerações ao escrever um modelo de lançamento personalizado pode ajudá-lo a criar um para uso com o AWS PCS. Para obter mais informações sobre modelos de execução, consulte Como iniciar uma instância a partir de um modelo de execução no Guia EC2 do usuário da Amazon.

Tópicos

Visão geral dos modelos de lançamento no AWS PCS

Há mais de 30 parâmetros disponíveis que você pode incluir em um modelo de EC2 execução, controlando muitos aspectos de como as instâncias são configuradas. A maioria é totalmente compatível com o AWS PCS, mas há algumas exceções.

Os seguintes parâmetros do modelo do EC2 Launch serão ignorados pelo AWS PCS, pois essas propriedades precisam ser gerenciadas diretamente pelo serviço:

- Tipo de instância/Especificar atributos do tipo de instância (`InstanceRequirements`) — O AWS PCS não oferece suporte à seleção de instância baseada em atributos.
- Tipo de instância (`InstanceType`) — Especifique os tipos de instância ao criar um grupo de nós.
- Detalhes avançados/Perfil da instância do IAM (`IamInstanceProfile`) — Você fornece isso ao criar ou atualizar o grupo de nós.

- Detalhes avançados/Desativar o encerramento da API (`DisableApiTermination`) — O AWS PCS deve controlar o ciclo de vida das instâncias do grupo de nós que ele executa.
- Detalhes avançados/Desativar API stop (`DisableApiStop`) — O AWS PCS deve controlar o ciclo de vida das instâncias do grupo de nós que ele executa.
- Detalhes avançados/Parar — Comportamento de hibernação (`HibernationOptions`) — O AWS PCS não suporta a hibernação de instâncias.
- Detalhes avançados/Elastic GPU () `ElasticGpuSpecifications` — A Amazon Elastic Graphics chegou ao fim da vida útil em 8 de janeiro de 2024.
- Detalhes avançados/Elastic Inference (`ElasticInferenceAccelerators`) — O Amazon Elastic Inference não está mais disponível para novos clientes.
- AAdvanced details/Specify CPU options/Threadspor núcleo (`ThreadsPerCore`) — O AWS PCS define o número de fios por núcleo como 1.

Esses parâmetros têm requisitos especiais que oferecem suporte à compatibilidade com o AWS PCS:

- Dados do usuário (`UserData`) — Isso deve ser codificado em várias partes. Consulte Trabalhando com dados de EC2 usuários da Amazon para AWS PCS.
- Imagens do aplicativo e do sistema operacional (`ImageId`) — Você pode incluir isso. No entanto, se você especificar uma ID de AMI ao criar ou atualizar o grupo de nós, ela substituirá o valor no modelo de execução. A AMI que você fornece deve ser compatível com o AWS PCS. Para obter mais informações, consulte "Amazon Machine Images (AMIs) para AWS PCS.
- Configurações de rede/Firewall (grupos de segurança) (**`SecurityGroups`**) — Uma lista de nomes de grupos de segurança não pode ser definida em um modelo de lançamento do AWS PCS. Você pode definir uma lista de grupos de segurança IDs (`SecurityGroupIds`), a menos que você defina interfaces de rede no modelo de execução. Em seguida, você deve especificar o grupo de segurança IDs para cada interface. Para obter mais informações, consulte Grupos de segurança no AWS PCS.
- Configurações de rede/Configuração de rede avançada (`NetworkInterfaces`) — Se você usa EC2 instâncias com uma única placa de rede e não exige nenhuma configuração de rede especializada, o AWS PCS pode configurar a rede de instâncias para você. Para configurar várias placas de rede ou habilitar o Elastic Fabric Adapter em suas instâncias, use`NetworkInterfaces`. Cada interface de rede deve ter uma lista de grupos de segurança IDs abaixo`Groups`. Para obter mais informações, consulte Várias interfaces de rede no AWS PCS.

- Detalhes avançados/reserva de capacidade (`CapacityReservationSpecification`) — Isso pode ser definido, mas não pode fazer referência a um específico `CapacityReservationId` ao trabalhar com AWS o PCS. No entanto, você pode referenciar um grupo de reserva de capacidade, onde esse grupo contém uma ou mais reservas de capacidade. Para obter mais informações, consulte Reservas de capacidade no AWS PCS.

Criar um modelo de execução básico

Você pode criar um modelo de lançamento usando o AWS Management Console ou AWS CLI o.

AWS Management Console

Para criar um modelo de execução

1. Abra o EC2console da Amazon e selecione Modelos de lançamento.
2. Escolha Criar modelo de execução.
3. Em Nome e descrição do modelo do Launch, insira um nome exclusivo e distinto para o nome do modelo do Launch.
4. Em Par de chaves (login) em Nome do par de chaves, selecione o par de chaves SSH que será usado para fazer login em EC2 instâncias gerenciadas pelo AWS PCS. Isso é opcional, mas recomendado.
5. Em Configurações de rede, depois em Firewall (grupos de segurança), escolha grupos de segurança a serem anexados à interface de rede. Todos os grupos de segurança no modelo de execução devem ser do seu cluster AWS PCS VPC. No mínimo, escolha:
   - Um grupo de segurança que permite a comunicação com o cluster AWS PCS
   - Um grupo de segurança que permite a comunicação entre EC2 instâncias iniciadas pelo AWS PCS
   - (Opcional) Um grupo de segurança que permite acesso SSH de entrada a instâncias interativas
   - (Opcional) Um grupo de segurança que permite que os nós de computação façam conexões de saída com a Internet
   - (Opcional) Grupos de segurança que permitem acesso a recursos em rede, como sistemas de arquivos compartilhados ou um servidor de banco de dados.

6. Seu novo ID do modelo de lançamento estará acessível no EC2 console da Amazon em Modelos de lançamento. O ID do modelo de lançamento terá o formulário`lt-0123456789abcdef01`.

Próxima etapa recomendada

- Use o novo modelo de execução para criar ou atualizar um grupo de nós de computação AWS PCS.

AWS CLI

Para criar um modelo de execução

Crie seu modelo de lançamento com o comando a seguir.

- Antes da execução do comando, realize as seguintes substituições:
  a. *region-code*Substitua pelo Região da AWS local em que você está trabalhando com o AWS PCS
  b. *my-launch-template-name*Substitua por um nome para seu modelo. Ele deve ser exclusivo do Conta da AWS e Região da AWS que você está usando.
  c. *my-ssh-key-name*Substitua pelo nome da sua chave SSH preferida.
  d. Substitua *sg-ExampleID1* e *sg-ExampleID2* por um grupo de segurança IDs que permite a comunicação entre suas EC2 instâncias e o agendador e a comunicação entre EC2 instâncias. Se você tiver apenas um grupo de segurança que habilite todo esse tráfego, poderá remover o `sg-ExampleID2` caractere de vírgula anterior. Você também pode adicionar mais grupos de segurança IDs. Todos os grupos de segurança que você inclui no modelo de execução devem ser do seu cluster AWS PCS 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 Isso exibirá um texto semelhante ao seguinte. O ID do modelo de lançamento é encontrado em`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"
    }
}
```

Próxima etapa recomendada

- Use o novo modelo de execução para criar ou atualizar um grupo de nós de computação AWS PCS.

Trabalhando com dados de EC2 usuários da Amazon para AWS PCS

Você pode fornecer dados EC2 do usuário em seu modelo de execução que `cloud-init` é executado quando suas instâncias são iniciadas. Os blocos de dados do usuário com o tipo de conteúdo são `cloud-config` executados antes do registro da instância na API AWS PCS, enquanto os blocos de dados do usuário com o tipo de conteúdo são `text/x-shellscript` executados após a conclusão do registro, mas antes do início do daemon do Slurm. Para obter mais informações sobre os tipos de conteúdo, consulte a documentação do cloud-init.

nossos dados de usuário podem realizar cenários de configuração comuns, incluindo, mas não se limitando ao seguinte:

- Incluindo usuários ou grupos
- Instalando pacotes
- Criação de partições e sistemas de arquivos
- Montagem de sistemas de arquivos de rede

Os dados do usuário nos modelos de lançamento devem estar no formato de arquivamento de várias partes MIME. Isso ocorre porque seus dados de usuário são mesclados com outros dados de usuário

do AWS PCS que são necessários para configurar nós em seu grupo de nós. Você pode combinar vários blocos de dados de usuário em um único arquivo MIME de várias partes.

Um arquivo em várias partes MIME consiste nos seguintes componentes:

- O tipo de conteúdo e a declaração de limite da parte: `Content-Type: multipart/mixed; boundary="==BOUNDARY=="`
- A declaração da versão MIME: `MIME-Version: 1.0`
- Um ou mais blocos de dados do usuário que contêm os seguintes componentes:
  - O limite de abertura, que sinaliza o início de um bloco de dados do usuário: `--==BOUNDARY==`. Você deve manter a linha antes desse limite em branco.
  - A declaração do tipo de conteúdo para o bloco: `Content-Type: text/cloud-config; charset="us-ascii"` ou`Content-Type: text/x-shellscript; charset="us-ascii"`. Você deve manter a linha após o branco da declaração do tipo de conteúdo.
  - O conteúdo de dados do usuário, por exemplo, uma lista de comandos de shell ou diretivas do `cloud-config`.
- O limite de fechamento que sinaliza o fim do arquivo MIME de várias partes: `--==BOUNDARY==--`. Você deve manter a linha antes do branco do limite de fechamento.

> **Note**
>
> Se você adicionar dados do usuário a um modelo de lançamento no EC2 console da Amazon, poderá colá-los como texto sem formatação. Ou você pode fazer o upload de um arquivo. Se você usar o AWS CLI ou um AWS SDK, primeiro deverá codificar os dados do usuário em base64 e enviar essa string como o valor do `UserData` parâmetro ao chamar CreateLaunchTemplate, conforme mostrado neste arquivo JSON.

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

Exemplos

- Exemplo: instalar software a partir de um repositório de pacotes
- Exemplo: executar scripts a partir de um bucket do S3
- Exemplo: definir variáveis de ambiente globais
- Usando sistemas de arquivos de rede com AWS PCS
- Exemplo: usar um sistema de arquivos EFS como um diretório inicial compartilhado

Exemplo: instalar software para AWS PCS a partir de um repositório de pacotes

Forneça esse script como valor de `"userData"` em seu modelo de lançamento. Para obter mais informações, consulte Trabalhando com dados de EC2 usuários da Amazon para AWS PCS.

Esse script usa cloud-config para instalar pacotes de software em instâncias de grupos de nós no lançamento. Para obter mais informações, consulte os formatos de dados do usuário na documentação do cloud-init. Este exemplo instala `curl` e. `llvm`

> **Note**
>
> Suas instâncias devem ser capazes de se conectar aos repositórios de pacotes configurados.

```
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==--
```

Exemplo: executar scripts adicionais para AWS PCS a partir de um bucket do S3

Forneça esse script como valor de `"userData"` em seu modelo de lançamento. Para obter mais informações, consulte Trabalhando com dados de EC2 usuários da Amazon para AWS PCS.

O script de dados do usuário a seguir usa cloud-config para importar um script de um bucket do S3 e executá-lo em instâncias de grupos de nós na inicialização. Para obter mais informações, consulte os formatos de dados do usuário na documentação do cloud-init.

Substitua os valores a seguir pelos seus próprios detalhes:

- *`amzn-s3-demo-bucket`*— O nome de um bucket do S3 que sua conta pode ler.
- *`object-key`*— A chave do objeto S3 do script a ser importado. Isso inclui o nome do script e sua localização na estrutura de pastas do bucket. Por exemplo, `scripts/script.sh`. Para obter mais informações, consulte Organização de objetos no console do Amazon S3 usando pastas no Guia do usuário do Amazon Simple Storage Service.
- *`shell`*— O shell Linux a ser usado para executar o script, como`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://amzn-s3-demo-bucket/object-key /tmp/script.sh
- /usr/bin/shell /tmp/script.sh

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

O perfil da instância do IAM para o grupo de nós deve ter acesso ao bucket. A política do IAM a seguir é um exemplo do bucket no script de dados do usuário acima.

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

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

Exemplo: definir variáveis de ambiente globais para AWS PCS

Forneça esse script como o valor de `"userData"` em seu modelo de lançamento. Para obter mais informações, consulte Trabalhando com dados de EC2 usuários da Amazon para AWS PCS.

O exemplo a seguir é usado `/etc/profile.d` para definir variáveis globais em instâncias de grupos de nós.

```
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==--
```

Exemplo: Use um sistema de arquivos EFS como um diretório inicial compartilhado para AWS PCS

Forneça esse script como valor de `"userData"` em seu modelo de lançamento. Para obter mais informações, consulte Trabalhando com dados de EC2 usuários da Amazon para AWS PCS.

Este exemplo estende o exemplo de montagem do EFS Usando sistemas de arquivos de rede com AWS PCS para implementar um diretório inicial compartilhado. O conteúdo de /home é copiado

antes da montagem do sistema de arquivos EFS. O conteúdo é então copiado rapidamente para o armazenamento compartilhado após a conclusão da montagem.

Substitua os seguintes valores nesse script pelos seus próprios detalhes:

- *`/mount-point-directory`*— O caminho em uma instância em que você deseja montar o sistema de arquivos EFS.
- *`filesystem-id`*— O ID do sistema de arquivos do sistema de arquivos EFS.

```
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==--
```

Exemplo: habilitar o SSH sem senha

Você pode usar o exemplo do diretório inicial compartilhado para implementar conexões SSH entre instâncias de cluster usando chaves SSH. Para cada usuário que usa o sistema de arquivos inicial compartilhado, execute um script semelhante ao seguinte:

```
#!/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**
>
> As instâncias devem usar um grupo de segurança que permita conexões SSH entre os nós do cluster.

Reservas de capacidade no AWS PCS

Você pode reservar a EC2 capacidade da Amazon em uma zona de disponibilidade específica e por um período específico usando reservas de capacidade sob demanda ou blocos de EC2 capacidade para garantir que você tenha a capacidade computacional necessária disponível quando precisar.

> **Note**
>
> AWS O PCS oferece suporte a reservas de capacidade sob demanda (ODCR), mas atualmente não oferece suporte a blocos de capacidade para ML.

Usando ODCRs com AWS PCS

Você pode escolher como o AWS PCS consome suas instâncias reservadas. Se você criar um ODCR aberto, todas as instâncias correspondentes iniciadas pelo AWS PCS ou outros processos em sua conta serão contabilizadas na reserva. Com um ODCR direcionado, somente as instâncias iniciadas com o ID de reserva específico são contabilizadas na reserva. Para cargas de trabalho urgentes, as segmentações ODCRs são mais comuns.

Você pode configurar um grupo de nós de computação AWS PCS para usar um ODCR direcionado adicionando-o a um modelo de execução. Aqui estão as etapas para fazer isso:

1. Crie uma reserva de capacidade sob demanda (ODCR) direcionada.
2. Adicione o ODCR a um grupo de reserva de capacidade.
3. Associe o grupo de reserva de capacidade a um modelo de lançamento.
4. Crie ou atualize um grupo de nós de computação AWS PCS para usar o modelo de lançamento.

Exemplo: reserve e use instâncias hpc6a.48xlarge com um ODCR direcionado

Esse exemplo de comando cria um ODCR direcionado para 32 instâncias hpc6a.48xlarge. Para iniciar as instâncias reservadas em um grupo de posicionamento, adicione `--placement-group-arn` ao comando. Você pode definir uma data de parada com `--end-date` e`--end-date-type`, caso contrário, a reserva continuará até ser encerrada manualmente.

```
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
```

O resultado desse comando será um ARN para o novo ODCR. Para usar o ODCR com o AWS PCS, ele deve ser adicionado a um grupo de reserva de capacidade. Isso ocorre porque o AWS PCS não oferece suporte individual ODCRs. Para obter mais informações, consulte Grupos de reserva de capacidade no Guia do usuário do Amazon Elastic Compute Cloud.

Veja como adicionar o ODCR a um grupo de reserva de capacidade chamado. `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
```

Com o ODCR criado e adicionado a um grupo de reserva de capacidade, agora ele pode ser conectado a um grupo de nós de computação do AWS PCS adicionando-o a um modelo de execução. Aqui está um exemplo de modelo de lançamento que faz referência ao grupo de reserva de capacidade.

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

Por fim, crie ou atualize um grupo de nós de computação AWS PCS para usar instâncias hpc6a.48xlarge e use o modelo de execução que faz referência ao ODCR em seu grupo de reserva de capacidade. Para um grupo de nós estático, defina instâncias mínima e máxima para o tamanho

da reserva (32). Para um grupo dinâmico de nós, defina as instâncias mínimas como 0 e as máximas até o tamanho da reserva.

Este exemplo é uma implementação simples de um único ODCR provisionado para um grupo de nós de computação. Mas, o AWS PCS suporta muitos outros designs. Por exemplo, você pode subdividir um grande grupo ODCR ou de reserva de capacidade entre vários grupos de nós de computação. Ou você pode usar ODCRs aquela outra conta da AWS criada e compartilhada com a sua. A principal restrição é que ODCRs sempre deve estar contida em um grupo de reserva de capacidade.

Para obter mais informações, consulte Reservas de capacidade sob demanda e blocos de capacidade para ML no Guia do usuário do Amazon Elastic Compute Cloud.

Parâmetros úteis do modelo de lançamento

Esta seção descreve alguns parâmetros do modelo de execução que podem ser amplamente úteis com o AWS PCS.

Ativar o CloudWatch monitoramento detalhado

Você pode ativar a coleta de CloudWatch métricas em um intervalo menor usando um parâmetro de modelo de lançamento.

AWS Management Console

Nas páginas do console para criar ou editar modelos de lançamento, essa opção é encontrada na seção Detalhes avançados. Defina CloudWatch Monitoramento detalhado como Ativar.

YAML

```
Monitoring:
    Enabled: True
```

JSON

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

Para obter mais informações, consulte Ativar ou desativar o monitoramento detalhado de suas instâncias no Guia do usuário do Amazon Elastic Compute Cloud para instâncias Linux.

Serviço de metadados de instância versão 2 (IMDS v2)

O uso do IMDS v2 com EC2 instâncias oferece aprimoramentos significativos de segurança e ajuda a reduzir os riscos potenciais associados ao acesso aos metadados da instância em ambientes. AWS

AWS Management Console

Nas páginas do console para criar ou editar modelos de lançamento, essa opção é encontrada na seção Detalhes avançados. Defina os metadados acessíveis como Ativados, a versão dos metadados somente como V2 (é necessário um token) e o limite de salto de resposta dos metadados como 4.

YAML

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

JSON

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

.

AWS Filas PCS

Uma fila AWS PCS é uma abstração leve sobre a implementação nativa de uma fila de trabalho do agendador. No caso do Slurm, uma fila AWS PCS é equivalente a uma partição do Slurm.

Os usuários enviam trabalhos para uma fila onde residem até que possam ser programados para execução em nós fornecidos por um ou mais grupos de nós de computação. Um cluster AWS PCS pode ter várias filas de trabalhos. Por exemplo, você pode criar uma fila que usa Amazon EC2 On-demand Instances para trabalhos de alta prioridade e outra fila que usa Amazon EC2 Spot Instances para trabalhos de baixa prioridade.

Tópicos

Criando uma fila no AWS PCS

Este tópico fornece uma visão geral das opções disponíveis e descreve o que considerar ao criar uma fila no AWS PCS.

Pré-requisitos

- Um cluster AWS PCS - as filas só podem ser criadas em associação com um cluster AWS PCS específico.
- Um ou mais grupos de nós de computação AWS PCS — uma fila deve estar associada a pelo menos um grupo de nós de computação AWS PCS.

Para criar uma fila no AWS PCS

Você pode criar uma fila usando o AWS Management Console ou o. AWS CLI

AWS Management Console

Para criar uma fila usando o console

1. Abra o console AWS PCS.

2. Selecione o cluster para a fila. Navegue até Filas e escolha Criar fila.
3. Na seção Configuração da fila, forneça os seguintes valores:

    a. Nome da fila — Um nome para sua fila. O nome só pode conter caracteres alfanuméricos (sensíveis a maiúsculas e minúsculas) e hifens. Ele deve começar com um caractere alfabético e não pode ter mais de 25 caracteres. O nome deve ser exclusivo dentro do cluster.

    b. Grupos de nós de computação — Selecione 1 ou mais grupos de nós de computação para atender a essa fila. Um grupo de nós de computação pode ser associado a mais de uma fila.
4. (Opcional) Em Tags, adicione quaisquer tags à sua fila AWS PCS
5. Selecione Criar fila. O campo Status mostrará Criando enquanto o AWS PCS cria a fila. A criação da fila pode levar vários minutos.

Próxima etapa recomendada

- Envie um trabalho para sua nova fila.

AWS CLI

Para criar uma fila usando AWS CLI

Use o comando a seguir para criar sua fila. Faça as seguintes substituições:

1. *`region-code`*Substitua pela AWS região do cluster. Por exemplo, `us-east-1`.
2. *`my-queue`*Substitua pelo nome da sua fila. O nome só pode conter caracteres alfanuméricos (sensíveis a maiúsculas e minúsculas) e hifens. Ele deve começar com um caractere alfabético e não pode ter mais de 25 caracteres. O nome deve ser exclusivo dentro do cluster.
3. *`my-cluster`*Substitua pelo nome ou ID do seu cluster.
4. *`compute-node-group-id`*Substitua pela ID do grupo de nós de computação para atender a fila. Por exemplo, `pcs_abcdef12345`.

> **Note**
>
> Ao criar uma fila, você deve fornecer a ID do grupo de nós de computação e não seu nome.

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

A criação da fila pode levar alguns minutos. Você pode consultar o status da sua fila com o comando a seguir. Você não poderá enviar trabalhos para a fila até que seu status chegue`ACTIVE`.

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

Próxima etapa recomendada

- Envie um trabalho para sua nova fila

Atualizando uma fila AWS PCS

Este tópico fornece uma visão geral das opções disponíveis e descreve o que considerar ao atualizar uma fila AWS PCS.

Considerações ao atualizar uma fila AWS PCS

As atualizações da fila não afetarão os trabalhos em execução, mas o cluster pode não conseguir aceitar novos trabalhos enquanto a fila estiver sendo atualizada.

Para atualizar uma fila AWS PCS

Você pode usar o AWS Management Console ou AWS CLI para atualizar uma fila.

AWS Management Console

Para atualizar uma fila

1. Abra o console AWS PCS em `https://console.aws.amazon.com/pcs/home#/clusters`
2. Selecione o cluster em que você deseja atualizar uma fila.
3. Navegue até Filas, vá até a fila que deseja atualizar e selecione Editar.
4. Na seção de configuração da fila, atualize qualquer um dos seguintes valores:
   - Grupos de nós — adicione ou remova grupos de nós de computação da associação com a fila.
   - Tags — Adicione ou remova tags da fila.
5. Selecione Atualizar. O campo Status mostrará Atualizando enquanto as alterações estão sendo aplicadas.

> **Important**
>
> As atualizações da fila podem levar vários minutos.

AWS CLI

Para atualizar uma fila

1. Atualize sua fila com o comando a seguir. Antes da execução do comando, realize as seguintes substituições:
   a. *`region-code`*Substitua por Região da AWS aquela em que você deseja criar seu cluster.
   b. *`my-queue`*Substitua pelo nome ou `computeNodeGroupId` pela sua fila.
   c. *`my-cluster`*Substitua pelo nome ou pelo nome `clusterId` do seu cluster.
   d. Para alterar as associações de grupos de nós de computação, forneça uma lista atualizada para`--compute-node-group-configurations`.
      - Por exemplo, para adicionar um segundo grupo `computeNodeGroupExampleID2` de nós de computação:

```
--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. A atualização da fila pode levar alguns minutos. Você pode consultar o status da sua fila com o comando a seguir. Você não poderá enviar trabalhos para a fila até que seu status chegue`ACTIVE`.

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

Próximas etapas recomendadas

- Envie um trabalho para sua fila atualizada.

Excluindo uma fila no PCS AWS

Este tópico fornece uma visão geral de como excluir uma fila no AWS PCS.

Considerações ao excluir uma fila

- Se houver trabalhos em execução na fila, eles serão encerrados pelo agendador quando a fila for excluída. Os trabalhos pendentes na fila serão cancelados. Considere esperar que os trabalhos na fila sejam concluídos ou interrompa/cancele manualmente usando os comandos nativos do agendador (como `scancel` para o Slurm).

Excluir a fila

Você pode usar o AWS Management Console ou AWS CLI para excluir uma fila.

AWS Management Console

Para excluir uma fila

1. Abra o console AWS PCS.
2. Selecione o cluster da fila.
3. Navegue até Filas e selecione a fila a ser excluída.
4. Escolha Excluir.
5. O campo Status é exibido`Deleting`. Pode demorar vários minutos para isso ser concluído.

> **Note**
>
> Você pode usar comandos nativos do seu agendador para confirmar que a fila foi excluída. Por exemplo, use `sinfo` ou `squeue` para o Slurm.

AWS CLI

Para excluir uma fila

- Use o comando a seguir para excluir uma fila, com essas substituições:
  - *`region-code`*Substitua por aquele em que Região da AWS seu cluster está.
  - *`my-queue`*Substitua pelo nome ou ID da sua fila.
  - *`my-cluster`*Substitua pelo nome ou ID do seu cluster.

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

Pode levar alguns minutos para excluir a fila.

> **Note**
>
> Você pode usar comandos nativos do seu agendador para confirmar que a fila foi excluída. Por exemplo, use `sinfo` ou `squeue` para o Slurm.

AWS nós de login do PCS

Um cluster AWS PCS geralmente precisa de pelo menos 1 nó de login para oferecer suporte ao acesso interativo e ao gerenciamento de tarefas. Uma forma de fazer isso é com um grupo estático de nós de computação AWS PCS configurado para a capacidade de nó de login. Você também pode configurar uma EC2 instância independente para atuar como um nó de login.

Tópicos

Usando um grupo de nós de computação AWS PCS para fornecer nós de login

Este tópico fornece uma visão geral das opções de configuração sugeridas e descreve o que considerar ao usar um grupo de nós de computação do AWS PCS para fornecer acesso persistente e interativo ao seu cluster.

Criação de um grupo de nós de computação AWS PCS para nós de login

Operacionalmente, isso não é muito diferente de criar um grupo normal de nós de computação. No entanto, existem algumas das principais opções de configuração que você pode fazer:

- Defina uma configuração de escalabilidade estática de pelo menos uma EC2 instância no grupo de nós de computação.
- Escolha a opção de compra sob demanda para evitar que suas instâncias sejam recuperadas.
- Escolha um nome informativo para o grupo de nós de computação, como login.
- Se você quiser que as instâncias do nó de login sejam acessíveis fora da sua VPC, considere usar uma sub-rede pública.
- Se você pretende permitir o acesso SSH, o modelo de lançamento precisará ter um grupo de segurança que exponha a porta SSH aos endereços IP de sua escolha.
- O perfil da instância do IAM deve ter somente as permissões da AWS que você deseja que seus usuários finais tenham. Para mais detalhes, consulte Perfis de instância do IAM para o AWS Parallel Computing Service.

- Considere permitir que o AWS Systems Manager Session Manager gerencie suas instâncias de login.
- Considere restringir o acesso às credenciais da AWS da instância somente para usuários administrativos.
- Selecione tipos de instância mais baratos do que para grupos de nós de computação comuns, pois os nós de login serão executados continuamente.
- Use a mesma AMI (ou uma derivada) dos outros grupos de nós de computação para ajudar a garantir que todas as instâncias tenham o mesmo software instalado. Para obter mais informações sobre personalização AMIs, consulte Amazon Machine Images (AMIs) para AWS PCS
- Configure as mesmas montagens do sistema de arquivos de rede (Amazon EFS, Amazon FSx for Lustre etc.) em seus nós de login e em suas instâncias de computação. Para obter mais informações, consulte Usando sistemas de arquivos de rede com AWS PCS.

Acesse seus nós de login

Quando seu novo grupo de nós de computação atingir o status ATIVO, você poderá encontrar as EC2 instâncias que ele criou e fazer login nelas. Para obter mais informações, consulte Encontrando instâncias de grupos de nós de computação no AWS PCS.

Atualização de um grupo de nós de computação AWS PCS para nós de login

Você pode atualizar um grupo de nós de login usando UpdateComputeNodeGroup o. Como parte do processo de atualização do grupo de nós, as instâncias em execução serão substituídas. Observe que isso interromperá todas as sessões ou processos ativos do usuário na instância. Os trabalhos do Slurm em execução ou em fila não serão afetados. Para obter mais informações, consulte Atualização de um grupo de nós de computação AWS PCS.

Você também pode editar o modelo de execução usado pelo seu grupo de nós de computação. Você deve usar UpdateComputeNodeGroup para aplicar o modelo de execução atualizado ao grupo de nós de computação. EC2 As novas instâncias lançadas no grupo de nós de computação usam o modelo de execução atualizado. Para obter mais informações, consulte Usando modelos de EC2 lançamento da Amazon com AWS PCS.

Excluindo um grupo de nós de computação AWS PCS para nós de login

Você pode atualizar um grupo de nós de login usando o mecanismo de exclusão de grupos de nós de computação no AWS PCS. As instâncias em execução serão encerradas como parte da exclusão do grupo de nós. Observe que isso interromperá todas as sessões ou processos ativos do usuário na instância. Os trabalhos do Slurm em execução ou em fila não serão afetados. Para obter mais informações, consulte Excluindo um grupo de nós de computação no PCS AWS.

Usando instâncias autônomas como nós de login do AWS PCS

Você pode configurar EC2 instâncias independentes para interagir com o agendador Slurm de um cluster AWS PCS. Isso é útil para criar nós de login, estações de trabalho ou hosts dedicados de gerenciamento de fluxo de trabalho que funcionam com clusters de AWS PCS, mas operam fora do gerenciamento de AWS PCS. Para fazer isso, cada instância autônoma deve:

1. Tenha uma versão compatível do software Slurm instalada.
2. Ser capaz de se conectar ao endpoint Slurmctld do cluster AWS PCS.
3. Configure adequadamente o Slurm Auth e o Cred Kiosk Daemon (`sackd`) com o endpoint e o segredo do cluster PCS. AWS Para obter mais informações, consulte sackd na documentação do Slurm.

Este tutorial ajuda você a configurar uma instância independente que se conecta a um cluster AWS PCS.

Sumário

Etapa 1 — Recupere o endereço e o segredo do cluster AWS PCS de destino

Recupere detalhes sobre o cluster AWS PCS de destino usando AWS CLI o comando a seguir. Antes da execução do comando, realize as seguintes substituições:

- *region-code*Substitua pelo Região da AWS local em que o cluster de destino está sendo executado.
- *cluster-ident*Substitua pelo nome ou identificador do cluster de destino

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

O comando retornará uma saída semelhante a este exemplo.

```
{
    "cluster": {
        "name": "get-started",
        "id": "pcs_123456abcd",
        "arn": "arn:aws:pcs:us-east-1:111122223333:cluster/pcs_123456abcd",
        "status": "ACTIVE",
        "createdAt": "2024-12-17T21:03:52+00:00",
        "modifiedAt": "2024-12-17T21:03:52+00:00",
        "scheduler": {
            "type": "SLURM",
            "version": "24.05"
        },
        "size": "SMALL",
        "slurmConfiguration": {
            "authKey": {
                "secretArn": "arn:aws:secretsmanager:us-east-1:111122223333:secret:pcs!
slurm-secret-pcs_123456abcd-a12ABC",
                "secretVersion": "ef232370-d3e7-434c-9a87-ec35c1987f75"
            }
        },
        "networking": {
            "subnetIds": [
                "subnet-0123456789abcdef0"
            ],
            "securityGroupIds": [
                "sg-0123456789abcdef0"
```

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

Neste exemplo, o endpoint do controlador Slurm do cluster tem um endereço IP de `10.3.149.220` e está sendo executado na porta. `6817` O `secretArn` será usado em etapas posteriores para recuperar o segredo do cluster. O endereço IP e a porta serão usados em etapas posteriores para configurar o `sackd` serviço.

Etapa 2 — Executar uma EC2 instância

Para iniciar uma EC2 instância

1. Abra o EC2 console da Amazon.
2. No painel de navegação, selecione Instances (Instâncias) e, depois, escolha Launch Instances (Iniciar instâncias) para abrir o novo assistente de inicialização de instância.
3. (Opcional) Na seção Nome e tags, forneça um nome para a instância, como`PCS-LoginNode`. O nome é atribuído à instância como uma etiqueta de recurso (`Name=PCS-LoginNode`).
4. Na seção Imagens do aplicativo e do sistema operacional, selecione uma AMI para um dos sistemas operacionais compatíveis com o AWS PCS. Para obter mais informações, consulte Sistemas operacionais compatíveis.
5. Na seção Tipo de instância, selecione um tipo de instância compatível. Para obter mais informações, consulte Tipos de instâncias compatíveis.
6. Na seção Par de chaves, selecione o par de chaves SSH a ser usado na instância.
7. Na seção Configurações de rede:
    - Escolha Editar.
        i. Selecione a VPC do seu cluster AWS PCS.
        ii. Em Firewall (grupos de segurança), escolha Selecionar grupo de segurança existente.

A. Selecione um grupo de segurança que permita o tráfego entre a instância e o controlador Slurm do cluster AWS PCS de destino. Para obter mais informações, consulte Requisitos e considerações do grupo de segurança.

B. (Opcional) Selecione um grupo de segurança que permita acesso SSH de entrada à sua instância.

8. Na seção Armazenamento, configure os volumes de armazenamento conforme necessário. Certifique-se de configurar espaço suficiente para instalar aplicativos e bibliotecas para habilitar seu caso de uso.
9. Em Avançado, escolha uma função do IAM que permita acesso ao segredo do cluster. Para obter mais informações, consulte Obtenha o segredo do cluster Slurm.
10. No painel Resumo, escolha Launch instance.

Etapa 3 — Instale o Slurm na instância

Quando a instância for iniciada e ficar ativa, conecte-se a ela usando seu mecanismo preferido. Use o instalador do Slurm fornecido por AWS para instalar o Slurm na instância. Para obter mais informações, consulte Instalador do Slurm.

Baixe o instalador do Slurm, descompacte-o e use o `installer.sh` script para instalar o Slurm. Para obter mais informações, consulte Etapa 3 — Instalar o Slurm.

Etapa 4 — Recuperar e armazenar o segredo do cluster

Essas instruções exigem AWS CLI o. Para obter mais informações, consulte Instalar ou atualizar para a versão mais recente do AWS CLI no Guia AWS Command Line Interface do Usuário da Versão 2.

Armazene o segredo do cluster com os comandos a seguir.

- Crie o diretório de configuração para o Slurm.

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

- Recupere, decodifique e armazene o segredo do cluster. Antes de executar esse comando, *`region-code`* substitua pela região em que o cluster de destino está sendo executado e *`secret-arn`* substitua pelo valor `secretArn` recuperado na Etapa 1.

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

> **Warning**
>
> Em um ambiente multiusuário, qualquer usuário com acesso à instância poderá obter o segredo do cluster se puder acessar o serviço de metadados da instância (IMDS). Isso, por sua vez, poderia permitir que eles se passassem por outros usuários. Considere restringir o acesso ao IMDS somente para usuários root ou administrativos. Como alternativa, considere usar um mecanismo diferente que não dependa do perfil da instância para buscar e configurar o segredo.

- Defina a propriedade e as permissões no arquivo de chave do Slurm.

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

> **Note**
>
> A chave Slurm deve pertencer ao usuário e ao grupo em que o `sackd` serviço é executado.

Etapa 5 — Configurar a conexão com o cluster AWS PCS

Para estabelecer uma conexão com o cluster AWS PCS, inicie `sackd` como um serviço do sistema seguindo estas etapas.

1. Configure o arquivo de ambiente para o `sackd` serviço com o comando a seguir. Antes de executar o comando, substitua *`ip-address`* e *`port`* pelos valores recuperados dos endpoints na Etapa 1.

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

2. Crie um arquivo `systemd` de serviço para gerenciar o `sackd` processo.

```
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-24.05/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. Defina a propriedade do arquivo `sackd` de serviço.

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

4. Ative o `sackd` serviço.

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

5. Inicie o serviço `sackd`.

```
sudo systemctl start sackd
```

Etapa 6 — (Opcional) Teste a conexão

Confirme se o `sackd` serviço está em execução. Segue um exemplo de saída. Se houver erros, eles geralmente aparecerão aqui.

```
[root@ip-10-3-27-112 ~]# systemctl status sackd
[x] sackd.service - Slurm auth and cred kiosk daemon
   Loaded: loaded (/etc/systemd/system/sackd.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2024-12-17 16:34:55 UTC; 8s ago
 Main PID: 9985 (sackd)
   CGroup: /system.slice/sackd.service
           ##9985 /opt/aws/pcs/scheduler/slurm-24.05/sbin/sackd --systemd --conf-
server=10.3.149.220:6817

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

Confirme se as conexões com o cluster estão funcionando usando os comandos do cliente Slurm, como e. `sinfo` `squeue` Aqui está um exemplo de saída de`sinfo`.

```
[root@ip-10-3-27-112 ~]# /opt/aws/pcs/scheduler/slurm-24.11/bin/sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
all up infinite 4 idle~ compute-[1-4]
```

Você também deve poder enviar trabalhos. Por exemplo, um comando semelhante a esse exemplo iniciaria um trabalho interativo em 1 nó no cluster.

```
/opt/aws/pcs/scheduler/slurm-24.11/bin/srun --nodes=1 -p all --pty bash -i
```

AWS Rede PCS

Seu cluster AWS PCS é criado em uma Amazon VPC. Este capítulo inclui os tópicos a seguir sobre redes para o agendador e os nós do seu cluster.

Com exceção da escolha de uma sub-rede para executar instâncias, você deve usar modelos de EC2 execução para configurar a rede para grupos de nós de computação do AWS PCS. Para obter mais informações sobre modelos de inicialização, consulte Usando modelos de EC2 lançamento da Amazon com AWS PCS.

Tópicos

AWS Requisitos e considerações sobre PCS, VPC e sub-rede

Ao criar um cluster AWS PCS, você especifica uma VPC e uma sub-rede nessa VPC. Este tópico fornece uma visão geral dos requisitos e considerações específicos do AWS PCS para a VPC e a (s) sub-rede (s) que você usa com seu cluster. Se você não tiver uma VPC para usar com o AWS PCS, poderá criar uma usando um modelo fornecido AWS. AWS CloudFormation Para obter mais informações sobre VPCs, consulte Nuvens privadas virtuais (VPC) no Guia do usuário da Amazon VPC.

Requisitos e considerações para VPCs

Quando você cria um cluster, a VPC especificada deve atender aos requisitos e considerações a seguir:

- A VPC deve ter um número suficiente de endereços IP disponíveis para o cluster, todos os nós e outros recursos de cluster que você deseja criar. Para obter mais informações, consulte o endereçamento IP para você VPCs e suas sub-redes no Guia do usuário da Amazon VPC.

- A VPC deve ter um nome de host DNS e suporte à resolução de DNS. Caso contrário, os nós não poderão registrar o cluster do cliente. Para ter mais informações, consulte Atributos de DNS para sua VPC no Guia do usuário da Amazon VPC.
- A VPC pode exigir o uso de VPC endpoints AWS PrivateLink para poder entrar em contato com a API PCS. AWS Para obter mais informações, consulte Conecte sua VPC aos serviços usados AWS PrivateLink no Guia do usuário da Amazon VPC.

> **Important**
>
> AWS O PCS não oferece suporte a uma VPC com locação de instância dedicada. A VPC que você usa para AWS PCS deve usar a locação de `default` instâncias. Você pode alterar a locação da instância para uma VPC existente. Para obter mais informações, consulte Alterar a locação da instância de uma VPC no Guia do usuário do Amazon Elastic Compute Cloud.

Requisitos e considerações para sub-redes

Quando você cria um cluster Slurm, o AWS PCS cria uma interface de rede elástica (ENI) na sub-rede especificada. Essa interface de rede permite a comunicação entre o controlador do agendador e a VPC do cliente. A interface de rede também permite que o Slurm se comunique com os componentes implantados na conta do cliente. Você só pode especificar a sub-rede de um cluster no momento da criação.

Requisitos de sub-redes para clusters

A sub-rede que você especifica ao criar um cluster deve atender aos seguintes requisitos:

- A sub-rede deve ter pelo menos 1 endereço IP para ser usada pelo AWS PCS.
- A sub-rede não pode residir em AWS Outposts AWS Wavelength, ou em uma zona AWS local.
- A sub-rede pode ser pública ou privada. Recomendamos que você especifique uma sub-rede privada, se possível. Uma sub-rede pública é uma sub-rede com uma tabela de rotas que inclui uma rota para um gateway da Internet; uma sub-rede privada é uma sub-rede com uma tabela de rotas que não inclui uma rota para um gateway da Internet.

Requisitos de sub-redes para nós

Você pode implantar nós e outros recursos de cluster na sub-rede especificada ao criar seu cluster AWS PCS e em outras sub-redes na mesma VPC.

Qualquer sub-rede na qual você implanta nós e recursos de cluster deve atender aos seguintes requisitos:

- Você deve garantir que a sub-rede tenha endereços IP disponíveis suficientes para implantar todos os nós e recursos do cluster.
- Se você planeja implantar nós em uma sub-rede pública, essa sub-rede deve atribuir endereços públicos automaticamente IPv4 .
- Se a sub-rede na qual você implanta nós for uma sub-rede privada e sua tabela de rotas não incluir uma rota para um dispositivo de tradução de endereços de rede (NAT) (), adicione endpoints de VPC usando a VPC do cliente. IPv4 AWS PrivateLink Os endpoints VPC são necessários para todos os AWS serviços com os quais os nós entram em contato. O único endpoint necessário é que o AWS PCS permita que o nó chame a ação da `RegisterComputeNodeGroupInstance` API. Para obter mais informações, consulte RegisterComputeNodeGroupInstancea Referência da API AWS PCS.
- O status da sub-rede pública ou privada não afeta o AWS PCS; os endpoints necessários devem estar acessíveis.

Criação de uma VPC para seu AWS cluster PCS

Você pode criar uma Amazon Virtual Private Cloud (Amazon VPC) para seus clusters dentro do AWS Parallel Computing Service (AWS PCS).

Use a Amazon VPC para lançar recursos de VPC em uma rede virtual que você definiu. Essa rede virtual é muito semelhante a uma rede tradicional que pode ser operada no seu próprio data center. Porém, ela vem com os benefícios do uso da infraestrutura escalável da Amazon Web Services. Recomendamos que você tenha uma compreensão completa do serviço Amazon VPC antes de implantar clusters VPC de produção. Para obter mais informações, consulte O que é Amazon VPC? no modo visual do autor. Guia do usuário do Amazon VPC.

Um cluster PCS, nós e recursos de suporte (como sistemas de arquivos e serviços de diretório) são implantados em sua Amazon VPC. Se você quiser usar uma Amazon VPC existente com o PCS, ela deverá atender aos requisitos descritos em. AWS Requisitos e considerações sobre PCS, VPC

e sub-rede Este tópico descreve como criar uma VPC que atenda aos requisitos de PCS usando um modelo fornecido AWS. AWS CloudFormation Depois de implantar um modelo, você pode visualizar os recursos criados por ele para saber exatamente quais recursos foram criados e a configuração desses recursos.

Pré-requisitos

Para criar uma Amazon VPC para PCS, você deve ter as permissões do IAM necessárias para criar recursos da Amazon VPC. Esses recursos são sub-redes VPCs, grupos de segurança, tabelas e rotas de rotas e gateways de internet e NAT. Para obter mais informações, consulte Criar uma VPC com uma sub-rede pública no Guia do usuário da Amazon VPC. Para revisar a lista completa da Amazon EC2, consulte Ações, recursos e chaves de condição da Amazon EC2 na Referência de autorização de serviço.

Crie uma Amazon VPC

Crie uma VPC copiando e colando a URL apropriada para Região da AWS onde você usará o PCS. Você também pode baixar o AWS CloudFormation modelo e enviá-lo você mesmo para o AWS CloudFormation console.

- Leste dos EUA (Norte da Virgínia) (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
```

- Leste dos EUA (Ohio) (us-east-2)

```
https://console.aws.amazon.com/cloudformation/home?region=us-east-2#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Oeste dos EUA (Oregon) (us-west-2)

```
https://console.aws.amazon.com/cloudformation/home?region=us-west-2#/stacks/
create/review?stackName=hpc-networking&templateURL=https://aws-hpc-recipes.s3.us-
east-1.amazonaws.com/main/recipes/net/hpc_large_scale/assets/main.yaml
```

- Somente modelo

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

Para criar uma Amazon VPC para PCS

1. Abra o modelo no AWS CloudFormation console.

   > **Note**
   >
   > Eles são pré-preenchidos no modelo para que você possa simplesmente deixá-los como valores padrão.

2. Em Forneça um nome de pilha, depois em Nome da pilha, insira. `hpc-networking`
3. Em parâmetros, insira os seguintes detalhes:

   a. Em VPC, em seguida, digite CidrBlock`10.3.0.0/16`

   b. Em Sub-redes A:

      i. Em seguida, CidrPublicSubnetA, insira `10.3.0.0/20`

      ii. Em seguida, CidrPrivateSubnetA, insira `10.3.128.0/20`

   c. Em Sub-redes B:

      i. Em seguida, CidrPublicSubnetB, insira `10.3.16.0/20`

      ii. Em seguida, CidrPrivateSubnetA, insira `10.3.144.0/20`

   d. Em Sub-redes C:

      i. Para ProvisionSubnetsC, selecione`True`.

      > **Note**
      >
      > Se você estiver criando uma VPC em uma região com menos de três zonas de disponibilidade, essa opção será ignorada se definida como. `True`

      ii. Em seguida, CidrPublicSubnetB, insira `10.3.32.0/20`

      iii. Em seguida, CidrPrivateSubnetA, insira `10.3.160.0/20`

4. Em Capacidades, marque a caixa Eu reconheço que a AWS CloudFormation pode criar recursos do IAM.

Monitore o status da AWS CloudFormation pilha. Quando chegar`CREATE_COMPLETE`, o recurso de VPC estará pronto para você usar.

> **Note**
>
> Para ver todos os recursos criados pelo AWS CloudFormation modelo, abra o AWS CloudFormation console. Escolha a pilha `hpc-networking` e depois a guia Resources (Recursos).

Grupos de segurança no AWS PCS

Os grupos de segurança na Amazon EC2 atuam como firewalls virtuais para controlar o tráfego de entrada e saída para as instâncias. Use um modelo de execução para um grupo de nós de computação do AWS PCS para adicionar ou remover grupos de segurança de suas instâncias. Se seu modelo de lançamento não contiver nenhuma interface de rede, use `SecurityGroupIds` para fornecer uma lista de grupos de segurança. Se seu modelo de execução definir interfaces de rede, você deverá usar o `Groups` parâmetro para atribuir grupos de segurança a cada interface de rede. Para obter mais informações sobre modelos de inicialização, consulte Usando modelos de EC2 lançamento da Amazon com AWS PCS.

> **Note**
>
> As alterações na configuração do grupo de segurança no modelo de execução afetam somente as novas instâncias lançadas após a atualização do grupo de nós de computação.

Requisitos e considerações do grupo de segurança

AWS O PCS cria uma interface de rede elástica (ENI) entre contas na sub-rede que você especifica ao criar um cluster. Isso fornece ao agendador de HPC, que está sendo executado em uma conta gerenciada por AWS, um caminho para se comunicar com EC2 instâncias lançadas pelo AWS PCS. Você deve fornecer um grupo de segurança para essa ENI que permita a comunicação bidirecional entre a ENI do agendador e suas instâncias de cluster. EC2

Uma maneira simples de fazer isso é criar um grupo de segurança autorreferenciado permissivo que permita o tráfego TCP/IP em todas as portas entre todos os membros do grupo. Você pode anexar isso tanto ao cluster quanto às EC2 instâncias do grupo de nós.

Exemplo de configuração permissiva de grupo de segurança

Tipo de regra	Protocolos	Portas	Origem	Destino
Entrada	Todos	Todos	Self	
Saída	Todos	Tudo		0.0.0.0/0
Saída	Todos	Todos		Self

Essas regras permitem que todo o tráfego flua livremente entre o controlador Slurm e os nós, permitem que todo o tráfego de saída chegue a qualquer destino e habilite o tráfego EFA.

Exemplo de configuração restritiva de grupo de segurança

Você também pode limitar as portas abertas entre o cluster e seus nós de computação. Para o agendador do Slurm, o grupo de segurança anexado ao seu cluster deve permitir as seguintes portas:

- 6817 — habilite conexões de entrada para instâncias de origem `slurmctld` EC2
- 6818 — habilite conexões de saída `slurmctld` para `slurmd` execução em instâncias EC2

O grupo de segurança conectado aos seus nós de computação deve permitir as seguintes portas:

- 6817 — habilite conexões de saída para instâncias `slurmctld` de EC2 origem.
- 6818 — habilite conexões de entrada e saída de e para `slurmd` instâncias de `slurmctld` grupos `slurmd` de nós
- 60001—63000 — conexões de entrada e saída entre instâncias de grupos de nós para oferecer suporte `srun`
- Tráfego EFA entre instâncias de grupos de nós. Para obter mais informações, consulte Preparar um grupo de segurança habilitado para EFA no Guia do usuário para instâncias Linux
- Qualquer outro tráfego entre nós exigido pela sua carga de trabalho

Várias interfaces de rede no AWS PCS

Algumas EC2 instâncias têm várias placas de rede. Isso permite que eles forneçam maior desempenho de rede, incluindo recursos de largura de banda acima de 100 Gbps e melhor manuseio de pacotes. Para obter mais informações sobre instâncias com várias placas de rede, consulte Interfaces de rede elásticas no Guia do usuário do Amazon Elastic Compute Cloud.

Configure placas de rede adicionais para instâncias em um grupo de nós de computação AWS PCS adicionando interfaces de rede ao modelo de EC2 execução. Abaixo está um exemplo de modelo de lançamento que permite duas placas de rede, como as encontradas em uma `hpc7a.96xlarge` instância. Observe os detalhes a seguir:

- A sub-rede de cada interface de rede deve ser a mesma que você escolheu ao configurar o grupo de nós de computação AWS PCS que usará o modelo de execução.
- O dispositivo de rede principal, onde ocorrerá a comunicação de rede de rotina, como tráfego SSH e HTTPS, é estabelecido definindo um `DeviceIndex` de`0`. Outras interfaces de rede têm um `DeviceIndex` de`1`. Só pode haver uma interface de rede primária — todas as outras interfaces são secundárias.
- Todas as interfaces de rede devem ter uma interface exclusiva`NetworkCardIndex`. Uma prática recomendada é numerá-los sequencialmente conforme definidos no modelo de lançamento.
- Os grupos de segurança para cada interface de rede são definidos usando`Groups`. Neste exemplo, um grupo de segurança SSH de entrada (`sg-`*`SshSecurityGroupId`*) é adicionado à interface de rede primária, bem como o grupo de segurança que permite comunicações dentro do cluster (). `sg-`*`ClusterSecurityGroupId`* Finalmente, um grupo de segurança que permite conexões de saída com a Internet (`sg-`*`InternetOutboundSecurityGroupId`*) é adicionado às interfaces primária e secundária.

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

Grupos de posicionamento para EC2 instâncias no AWS PCS

Você pode usar um grupo de posicionamento para influenciar o posicionamento das EC2 instâncias de acordo com as necessidades da carga de trabalho que é executada nelas.

Tipos de grupos de posicionamento

- Cluster — agrupa as instâncias em uma zona de disponibilidade para otimizar a comunicação de baixa latência.
- Partição — distribui instâncias em partições lógicas para ajudar a maximizar a resiliência.
- Spread — impõe rigorosamente que um pequeno número de instâncias seja executado em hardware distinto, o que também pode ajudar na resiliência.

Para obter mais informações, consulte Grupos de posicionamento para suas EC2 instâncias da Amazon no Guia do usuário do Amazon Elastic Compute Cloud.

Recomendamos que você inclua um grupo de posicionamento de cluster ao configurar um grupo de nós de computação AWS PCS para usar o Elastic Fabric Adapter (EFA).

Para criar um grupo de posicionamento de clusters que funcione com o EFA

1. Crie um grupo de posicionamento com o tipo de cluster para o grupo de nós de computação.
   - Use o seguinte AWS CLI comando:

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

   - Você também pode usar um CloudFormation modelo para criar um grupo de posicionamento. Para obter mais informações, consulte Como trabalhar com CloudFormation modelos no Guia

AWS CloudFormation do usuário. Faça o download do modelo a partir do URL a seguir e faça o upload para o CloudFormation console.

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

2. Inclua o grupo de posicionamento no modelo de EC2 lançamento do grupo de nós de computação AWS PCS.

Usando o Elastic Fabric Adapter (EFA) com PCS AWS

O Elastic Fabric Adapter (EFA) é uma interconexão de rede avançada de alto desempenho AWS que você pode conectar à sua EC2 instância para acelerar aplicativos de computação de alta performance (HPC) e aprendizado de máquina. Habilitar seus aplicativos em execução em um cluster AWS PCS com o EFA envolve a configuração das instâncias do grupo de nós de computação do AWS PCS para usar o EFA da seguinte maneira.

> **Note**
>
> Instale o EFA em uma AMI AWS compatível com PC — A AMI usada no grupo de nós de computação AWS do PCS deve ter o driver EFA instalado e carregado. Para obter informações sobre como criar uma AMI personalizada com o software EFA instalado, consulteImagens personalizadas da Amazon Machine (AMIs) para AWS PCS.

Sumário

Identifique instâncias habilitadas para EFA EC2

Para usar o EFA, todos os tipos de instância permitidos para um grupo de computação AWS PCS devem oferecer suporte ao EFA e devem ter o mesmo número de v CPUs (e GPUs se apropriado). Para obter uma lista de instâncias habilitadas para EFA, consulte o Elastic Fabric Adapter para cargas de trabalho de HPC e ML na Amazon EC2 no Guia do usuário do Amazon Elastic Compute Cloud. Você também pode usar o AWS CLI para ver uma lista de tipos de instância compatíveis com o EFA. *region-code*Substitua pelo Região da AWS local em que você usa o AWS PCS, como`us-east-1`.

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

> **Note**
>
> Determine quantas interfaces de rede estão disponíveis — Algumas EC2 instâncias têm várias placas de rede. Isso permite que eles tenham vários EFAs. Para obter mais informações, consulte Várias interfaces de rede no AWS PCS.

Crie um grupo de segurança para apoiar as comunicações da EFA

AWS CLI

Você pode usar o AWS CLI comando a seguir para criar um grupo de segurança que ofereça suporte ao EFA. O comando gera um ID de grupo de segurança. Faça as seguintes substituições:

- *region-code*— Especifique Região da AWS onde você usa o AWS PCS, como`us-east-1`.
- *vpc-id*— Especifique o ID da VPC que você usa para AWS PCS.
- *efa-group-name*— Forneça o nome escolhido para o grupo de segurança.

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

```
    --vpc-id vpc-id \
    --region region-code
```

Use os comandos a seguir para anexar regras de grupos de segurança de entrada e saída. Faça a seguinte substituição:

- *efa-secgroup-id*— Forneça o ID do grupo de segurança EFA que você acabou de criar.

```
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

Você pode usar um CloudFormation modelo para criar um grupo de segurança que ofereça suporte ao EFA. Faça o download do modelo a partir do URL a seguir e, em seguida, carregue-o no AWS CloudFormation console.

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

Com o modelo aberto no AWS CloudFormation console, insira as seguintes opções.

- Em Forneça um nome de pilha
  - Em Nome da pilha, insira um nome como`efa-sg-stack`.
- Em Parâmetros
  - Em SecurityGroupName, insira um nome como`efa-sg`.
  - Em VPC, selecione a VPC em que você usará o PCS. AWS

Conclua a criação da CloudFormation pilha e monitore seu status. Quando chega, `CREATE_COMPLETE` o grupo de segurança EFA está pronto para uso.

(Opcional) Crie um grupo de colocação

Recomendamos que você execute todas as instâncias que usam o EFA em um grupo de posicionamento de cluster para minimizar a distância física entre elas. Crie um grupo de posicionamento para cada grupo de nós de computação em que você planeja usar o EFA. Consulte Grupos de posicionamento para EC2 instâncias no AWS PCS para criar um grupo de posicionamento para seu grupo de nós de computação.

Criar ou atualizar um modelo de EC2 lançamento

As interfaces de rede EFA são configuradas no modelo de EC2 lançamento para um grupo de nós de computação AWS PCS. Se houver várias placas de rede, várias EFAs podem ser configuradas. O grupo de segurança EFA e o grupo de posicionamento opcional também estão incluídos no modelo de lançamento.

Aqui está um exemplo de modelo de lançamento para instâncias com duas placas de rede, como hpc7a.96xlarge. As instâncias serão lançadas `subnet-`*`SubnetID1`* em um grupo de posicionamento de clusterspg-*`PlacementGroupId1`*.

Grupos de segurança devem ser adicionados especificamente a cada interface EFA. Todo EFA precisa do grupo de segurança que habilita o tráfego do EFA ()`sg-`*`EfaSecGroupId`*. Outros grupos de segurança, especialmente aqueles que lidam com tráfego regular, como SSH ou HTTPS, só precisam estar conectados à interface de rede primária (designada por um `DeviceIndex` de`0`). Os modelos de inicialização em que as interfaces de rede são definidas não oferecem suporte à configuração de grupos de segurança usando o `SecurityGroupIds` parâmetro — você deve definir um valor para `Groups` cada interface de rede configurada.

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

Crie ou atualize grupos de nós de computação para o EFA

Seus grupos de nós de computação do AWS PCS devem conter instâncias que tenham o mesmo número de vCPUs, arquitetura de processador e suporte a EFA. Configure o grupo de nós de computação para usar a AMI com o software EFA instalado nela e para usar o modelo de lançamento que configura as interfaces de rede habilitadas para EFA.

(Opcional) Teste EFA

Você pode demonstrar a comunicação habilitada para EFA entre dois nós em um grupo de nós de computação executando o `fi_pingpong` programa, que está incluído na instalação do software EFA. Se esse teste for bem-sucedido, é provável que o EFA esteja configurado corretamente.

Para começar, você precisa de duas instâncias em execução no grupo de nós de computação. Se seu grupo de nós de computação usa capacidade estática, já deve haver instâncias disponíveis. Para um grupo de nós de computação que usa capacidade dinâmica, você pode iniciar dois nós usando o `salloc` comando. Aqui está um exemplo de um cluster com um grupo dinâmico de nós chamado `hpc7g` associado a uma fila chamada`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
```

Descubra o endereço IP dos dois nós alocados usando`scontrol`. No exemplo a seguir, os endereços são `10.3.140.69` para `hpc7g-1` e `10.3.132.211` para`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=24.11.5
   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=24.11.5
   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
```

Conecte-se a um dos nós (neste caso de exemplo`hpc7g-1`) usando SSH (ou SSM). Observe que esse é um endereço IP interno, portanto, talvez seja necessário se conectar a partir de um dos seus nós de login se usar SSH. Lembre-se também de que a instância precisa ser configurada com uma chave SSH por meio do modelo de execução do grupo de nós de computação.

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

Agora, inicie `fi_pingpong` no modo servidor.

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

Conecte-se à segunda instância (`hpc7g-2`).

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

Execute `fi_pingpong` no modo cliente, conectando-se ao servidor ativado`hpc7g-1`. Você deve ver uma saída semelhante ao exemplo abaixo.

```
% /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
```

(Opcional) Use um CloudFormation modelo para criar um modelo de lançamento habilitado para EFA

Como há várias dependências na configuração do EFA, foi fornecido um CloudFormation modelo que você pode usar para configurar um grupo de nós de computação. Ele suporta instâncias com até quatro placas de rede. Para saber mais sobre instâncias com várias placas de rede, consulte Interfaces de rede elásticas no Guia do usuário do Amazon Elastic Compute Cloud.

Faça o download do CloudFormation modelo a partir do seguinte URL e, em seguida, carregue-o no CloudFormation console em Região da AWS que você usa o AWS PCS.

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

Com o modelo aberto no AWS CloudFormation console, insira os valores a seguir. Observe que o modelo fornecerá alguns valores de parâmetros padrão. Você pode deixá-los como valores padrão.

- Em Forneça um nome de pilha
  - Em Nome da pilha, insira um nome descritivo. Recomendamos incorporar o nome que você escolherá para seu grupo de nós de computação do AWS PCS, como. *NODEGROUPNAME*`-efa-lt`
- Em Parâmetros
  - Em NumberOfNetworkCards, escolha o número de placas de rede nas instâncias que estarão em seu grupo de nós.
  - Em VpcId, escolha a VPC em que seu cluster AWS PCS está implantado.
  - Em NodeGroupSubnetId, escolha a sub-rede em seu cluster VPC onde as instâncias habilitadas para EFA serão executadas.
  - Em PlacementGroupName, deixe o campo em branco para criar um novo grupo de posicionamento de cluster para o grupo de nós. Se você tem um grupo de posicionamento existente que deseja usar, insira o nome dele aqui.
  - Em ClusterSecurityGroupId, escolha o grupo de segurança que você está usando para permitir o acesso a outras instâncias no cluster e à API AWS PCS. Muitos clientes escolhem o grupo de segurança padrão em seu cluster VPC.
  - Em SshSecurityGroupId, forneça o ID de um grupo de segurança que você está usando para permitir acesso SSH de entrada aos nós em seu cluster.
  - Para SshKeyName, selecione o par de chaves SSH para acessar os nós em seu cluster.
  - Para LaunchTemplateName, insira um nome descritivo para o modelo de lançamento, como*NODEGROUPNAME*`-efa-lt`. O nome deve ser exclusivo para você Conta da AWS no Região da AWS local em que você usará o AWS PCS.
- Em Capacidades
  - Marque a caixa “Eu reconheço que isso AWS CloudFormation pode criar recursos do IAM”.

Monitore o status da CloudFormation pilha. Quando chega, `CREATE_COMPLETE` o modelo de lançamento está pronto para ser usado. Use-o com um grupo de nós de computação AWS PCS, conforme descrito acima emCrie ou atualize grupos de nós de computação para o EFA.

Usando sistemas de arquivos de rede com AWS PCS

Você pode conectar sistemas de arquivos de rede a nós lançados em um grupo de nós de computação do Serviço de Computação AWS Paralela (AWS PCS) para fornecer um local persistente em que dados e arquivos possam ser gravados e acessados. Você pode usar sistemas de arquivos fornecidos por AWS serviços, incluindo Amazon Elastic File System (Amazon EFS), Amazon FSx for Lustre, Amazon FSx for NetApp ONTAP, Amazon FSx for OpenZFS e Amazon File Cache. Você também pode usar sistemas de arquivos autogerenciados, como servidores NFS.

Este tópico aborda considerações e exemplos do uso de sistemas de arquivos de rede com AWS PCS.

Considerações sobre o uso de sistemas de arquivos de rede

Os detalhes da implementação de vários sistemas de arquivos são diferentes, mas há algumas considerações comuns.

- O software do sistema de arquivos relevante deve estar instalado na instância. Por exemplo, para usar o Amazon FSx for Lustre, o apropriado Lustre o pacote deve estar presente. Isso pode ser feito incluindo-o no grupo de nós de computação AMI ou usando um script executado na inicialização da instância.
- Deve haver uma rota de rede entre o sistema de arquivos de rede compartilhado e as instâncias do grupo de nós de computação.
- As regras do grupo de segurança para o sistema de arquivos de rede compartilhado e as instâncias do grupo de nós de computação devem permitir conexões com as portas relevantes.
- Você deve manter uma consistência POSIX namespace de usuário e grupo entre os recursos que acessam os sistemas de arquivos. Caso contrário, trabalhos e processos interativos executados em seu cluster PCS poderão encontrar erros de permissão.
- As montagens do sistema de arquivos são feitas usando EC2 modelos de lançamento. Erros ou tempos limite na montagem de um sistema de arquivos de rede podem impedir que as instâncias se tornem disponíveis para executar trabalhos. Isso, por sua vez, pode levar a custos inesperados. Para obter mais informações sobre a depuração de modelos de lançamento, consulte. Usando modelos de EC2 lançamento da Amazon com AWS PCS

Exemplo de montagens de rede

Você pode criar sistemas de arquivos usando o Amazon EFS, o Amazon FSx for Lustre, o Amazon FSx for NetApp ONTAP, o Amazon FSx for OpenZFS e o Amazon File Cache. Expanda a seção relevante abaixo para ver um exemplo de cada montagem de rede.

Amazon EFS

Configuração do sistema de arquivos

Criar um sistema de arquivos do Amazon EFS. Certifique-se de que ele tenha um destino de montagem em cada zona de disponibilidade em que você iniciará as instâncias do grupo de nós de computação do PCS. Além disso, certifique-se de que cada destino de montagem esteja associado a um grupo de segurança que permita acesso de entrada e saída das instâncias do grupo de nós de computação do PCS. Para obter mais informações, consulte Montar alvos e grupos de segurança no Guia do usuário do Amazon Elastic File System.

Modelo de execução

Adicione os grupos de segurança da configuração do sistema de arquivos ao modelo de execução que você usará para o grupo de nós de computação.

Inclua dados do usuário que usam o `cloud-config` mecanismo para montar o sistema de arquivos Amazon EFS. Substitua os seguintes valores nesse script pelos seus próprios detalhes:

- *`mount-point-directory`*— O caminho em cada instância em que você montará o Amazon EFS
- *`filesystem-id`*— O ID do sistema de arquivos do sistema de arquivos EFS

```
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 para Lustre

Configuração do sistema de arquivos

Crie um sistema de arquivos FSx for Lustre na VPC onde você usará AWS o PCS. Para minimizar as transferências entre zonas, implante em uma sub-rede na mesma zona de disponibilidade em que você iniciará a maioria das instâncias do grupo de nós de computação do PCS. Certifique-se de que o sistema de arquivos esteja associado a um grupo de segurança que permita acesso de entrada e saída das instâncias do grupo de nós de computação do PCS. Para obter mais informações sobre grupos de segurança, consulte Controle de acesso ao sistema de arquivos com o Amazon VPC no Guia do usuário do Amazon FSx for Lustre.

Modelo de execução

Inclua dados do usuário usados `cloud-config` para montar o sistema de arquivos FSx for Lustre. Substitua os seguintes valores nesse script pelos seus próprios detalhes:

- *`mount-point-directory`*— O caminho em uma instância em que você deseja montar FSx para o Lustre
- *`filesystem-id`*— O ID do sistema de arquivos do sistema de arquivos FSx for Lustre
- *`mount-name`*— O nome da montagem do sistema de arquivos FSx for Lustre
- *`region-code`*— Região da AWS Onde o sistema de arquivos FSx for Lustre é implantado (deve ser o mesmo do seu sistema AWS PCS)
- (Opcional) *`latest`* — Qualquer versão do Lustre suportado por FSx for Lustre

```
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 para NetApp ONTAP

Configuração do sistema de arquivos

Crie um sistema de arquivos Amazon FSx for NetApp ONTAP na VPC onde você usará AWS o PCS. Para minimizar as transferências entre zonas, implante em uma sub-rede na mesma zona de disponibilidade em que você iniciará a maioria das instâncias do grupo de nós de computação do AWS PCS. Certifique-se de que o sistema de arquivos esteja associado a um grupo de segurança que permita acesso de entrada e saída das instâncias do grupo de nós de computação do AWS PCS. Para obter mais informações sobre grupos de segurança, consulte Controle de acesso ao sistema de arquivos com Amazon VPC no Guia do usuário do FSx ONTAP.

Modelo de execução

Inclua dados do usuário usados `cloud-config` para montar o volume raiz de um sistema de arquivos FSx for ONTAP. Substitua os seguintes valores nesse script pelos seus próprios detalhes:

- *`mount-point-directory`*— O caminho em uma instância em que você deseja montar seu volume FSx for ONTAP
- *`svm-id`*— O ID SVM FSx para o sistema de arquivos ONTAP
- *`filesystem-id`*— O ID do sistema de arquivos FSx para o sistema de arquivos ONTAP
- *`region-code`*— Região da AWS Onde o sistema de arquivos FSx for ONTAP está implantado (deve ser o mesmo do seu sistema AWS PCS)
- *`volume-name`*— O nome do volume FSx for ONTAP

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

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

runcmd:
- mkdir -p /mount-point-directory
```

```
- mount -t nfs svm-id.filesystem-id.fsx.region-code.amazonaws.com:/volume-name /mount-point-directory

--==MYBOUNDARY==
```

Amazon FSx para OpenZFS

Configuração do sistema de arquivos

Crie um sistema de arquivos FSx para OpenZFS na VPC onde você usará o PCS. AWS
Para minimizar as transferências entre zonas, implante em uma sub-rede na mesma zona de disponibilidade em que você iniciará a maioria das instâncias do grupo de nós de computação do AWS PCS. Certifique-se de que o sistema de arquivos esteja associado a um grupo de segurança que permita acesso de entrada e saída das instâncias do grupo de nós de computação do AWS PCS. Para obter mais informações sobre grupos de segurança, consulte Gerenciando o acesso ao sistema de arquivos com a Amazon VPC no Guia do usuário do FSx OpenZFS.

Modelo de execução

Inclua dados do usuário que são usados `cloud-config` para montar o volume raiz de um sistema de arquivos FSx para OpenZFS. Substitua os seguintes valores nesse script pelos seus próprios detalhes:

- *`mount-point-directory`*— O caminho em uma instância em que você deseja montar seu compartilhamento FSx para OpenZFS
- *`filesystem-id`*— O ID do sistema de arquivos FSx para o sistema de arquivos OpenZFS
- *`region-code`*— Região da AWS Onde o sistema de arquivos FSx for OpenZFS está implantado (deve ser o mesmo do seu AWS sistema PCS)

```
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 File Cache

Configuração do sistema de arquivos

Crie um Amazon File Cache na VPC onde você AWS usará o PCS. Para minimizar as transferências entre zonas, escolha uma sub-rede na mesma zona de disponibilidade em que você iniciará a maioria das instâncias do grupo de nós de computação do PCS. Verifique se o cache de arquivos está associado a um grupo de segurança que permite tráfego de entrada e saída na porta 988 entre suas instâncias do PCS e o cache de arquivos. Para obter mais informações sobre grupos de segurança, consulte Controle de acesso ao cache com o Amazon VPC no Guia do usuário do Amazon File Cache.

Modelo de execução

Adicione os grupos de segurança da configuração do sistema de arquivos ao modelo de execução que você usará para o grupo de nós de computação.

Inclua dados do usuário usados `cloud-config` para montar o Amazon File Cache. Substitua os seguintes valores nesse script pelos seus próprios detalhes:

- *`mount-point-directory`*— O caminho em uma instância em que você deseja montar FSx para o Lustre
- *`cache-dns-name`*— O nome do Sistema de Nomes de Domínio (DNS) para o cache de arquivos
- *`mount-name`*— O nome da montagem do cache de arquivos

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

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

runcmd:
- amazon-linux-extras install -y lustre=2.12
- mkdir -p /mount-point-directory
- mount -t lustre -o relatime,flock cache-dns-name@tcp:/mount-name /mount-point-
directory
```

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

Amazon Machine Images (AMIs) para AWS PCS

AWS O PCS trabalha com o AMIs que você fornece, oferecendo grande flexibilidade no software e na configuração encontrados nos nós do seu cluster. Se você estiver testando o AWS PCS, poderá usar uma amostra de AMI fornecida e mantida pela AWS. Se você estiver usando o AWS PCS na produção, recomendamos que você crie o seu próprio AMIs. Este tópico aborda como descobrir e usar a amostra AMIs, bem como criar e usar sua própria amostra personalizada AMIs.

Tópicos

Usando amostras de Amazon Machine Images (AMIs) com AWS PCS

A AWS fornece uma amostra AMIs que você pode usar como ponto de partida para trabalhar com o AWS PCS.

> **Important**
>
> AMIs As amostras são para fins de demonstração e não são recomendadas para cargas de trabalho de produção.

Encontre a amostra atual do AWS PCS AMIs

AWS Management Console

A amostra do AWS PCS AMIs tem a seguinte convenção de nomenclatura:

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

Valores aceitos

- *OS* – amzn2
- *architecture*: x86_64 ou arm64
- *scheduler* – slurm
- *scheduler-major-version* – 24.11

Para encontrar uma amostra de AWS PCS AMIs

1. Abra o EC2 console da Amazon.
2. Acesse AMIs.
3. Escolha Imagens públicas.
4. Em Localizar AMI por atributo ou tag, pesquise uma AMI usando o nome do modelo.

   Exemplos

   - Exemplo de AMI para Slurm 24.11 em instâncias Arm64

     ```
     aws-pcs-sample_ami-amzn2-arm64-slurm-24.11
     ```

   - Exemplo de AMI para Slurm 24.11 em instâncias x86

     ```
     aws-pcs-sample_ami-amzn2-x86_64-slurm-24.11
     ```

   > **Note**
   >
   > Se houver várias AMIs, use a AMI com o carimbo de data/hora mais recente.

5. Use o ID da AMI ao criar ou atualizar um grupo de nós de computação.

AWS CLI

Você pode encontrar o exemplo de AMI de AWS PCS mais recente com os comandos a seguir. *region-code*Substitua pelo Região da AWS local em que você usa o AWS PCS, comous-east-1.

- x86_64

```
aws ec2 describe-images --region region-code --owners amazon \
--filters 'Name=name,Values=aws-pcs-sample_ami-amzn2-x86_64-slurm-24.11*' \
            'Name=state,Values=available' \
--query 'sort_by(Images, &CreationDate)[-1].[Name,ImageId]' --output text
```

- Arm 64

```
aws ec2 describe-images --region region-code --owners amazon \
--filters 'Name=name,Values=aws-pcs-sample_ami-amzn2-arm64-slurm-24.11*' \
            'Name=state,Values=available' \
--query 'sort_by(Images, &CreationDate)[-1].[Name,ImageId]' --output text
```

Use o ID da AMI ao criar ou atualizar um grupo de nós de computação.

Saiba mais sobre a amostra AWS PCS AMIs

Para ver o conteúdo e os detalhes de configuração das versões atuais e anteriores da amostra AWS PCS AMIs, consulteNotas de lançamento da amostra AWS PCS AMIs.

Crie seu próprio AMIs compatível com AWS PCS

Para saber como criar seus próprios AMIs que funcionem com o AWS PCS, consulteImagens personalizadas da Amazon Machine (AMIs) para AWS PCS.

Imagens personalizadas da Amazon Machine (AMIs) para AWS PCS

AWS O PCS foi projetado para funcionar com Amazon Machine Images (AMI) que você traz para o serviço. Eles AMIs podem ter software e configurações arbitrários instalados neles, desde que tenham o agente AWS PCS e uma versão compatível do Slurm instalados e configurados corretamente. Você deve usar os AWS instaladores fornecidos para instalar o software AWS PCS em sua AMI personalizada. Recomendamos que você use AWS instaladores fornecidos para instalar o Slurm em sua AMI personalizada, mas você pode instalar o Slurm sozinho se preferir (não recomendado).

> **Note**
>
> Se quiser experimentar o AWS PCS sem criar uma AMI personalizada, você pode usar uma amostra de AMI fornecida pela AWS. Para obter mais informações, consulte Usando amostras de Amazon Machine Images (AMIs) com AWS PCS.

Este tutorial ajuda você a criar uma AMI que pode ser usada com grupos de nós de computação do PCS para potencializar suas cargas de trabalho de HPC e AI/ML.

Tópicos

Etapa 1 — Executar uma instância temporária

Execute uma instância temporária que você possa usar para instalar e configurar o software AWS PCS e o agendador Slurm. Você usa essa instância para criar uma AMI compatível com AWS PCS.

Para executar uma instância temporária

1. Abra o EC2 console da Amazon.
2. No painel de navegação, escolha Instâncias e, em seguida, escolha Launch instances para abrir o novo assistente de instância de inicialização.
3. (Opcional) Na seção Nome e tags, forneça um nome para a instância, como`PCS-AMI-instance`. O nome é atribuído à instância como uma etiqueta de recurso (`Name=PCS-AMI-instance`).
4. Na seção Application and OS Images (Imagens de aplicação e sistema operacional), selecione uma AMI para um dos sistemas operacionais compatíveis.
5. Na seção Instance type (Tipo de instância), selecione um tipo de instância compatível.

6. Na seção Key pair (Par de chaves), selecione o par de chaves a ser usado na instância.
7. Na seção Configurações de rede:
    - Para Firewall (grupos de segurança), escolha Selecionar grupo de segurança existente e, em seguida, selecione um grupo de segurança que permita acesso SSH de entrada à sua instância.
8. Na seção Storage (Armazenamento), configure os volumes conforme necessário. Certifique-se de configurar espaço suficiente para instalar seus próprios aplicativos e bibliotecas.
9. No painel Resumo painel, escolha Iniciar instância.

Etapa 2 — Instalar o agente AWS PCS

Instale o agente que configura as instâncias iniciadas pelo AWS PCS para uso com o Slurm. Para obter mais informações sobre o agente AWS PCS, consulteAWS Versões do agente PCS.

Para instalar o agente AWS PCS

1. Conecte à instância que você iniciou. Para obter mais informações, consulte Conectar-se à instância do Linux.
2. (Opcional) Para garantir que todos os seus pacotes de software estejam atualizados, faça uma rápida atualização de software na sua instância. esse processo pode demorar alguns minutos.
    - Amazon Linux 2, RHEL 9, Rocky Linux 9

      ```
      sudo yum update -y
      ```

    - Ubuntu 22.04

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

3. Reinicialize a instância e reconecte-se a ela.
4. Baixe os arquivos de instalação do agente AWS PCS. Os arquivos de instalação são empacotados em um arquivo tarball () `.tar.gz` compactado. Para fazer download da última versão estável, use o seguinte comando: *`region`*Substitua pelo Região da AWS local em que você iniciou sua instância temporária, como`us-east-1`.

```
curl https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.2.1-1.tar.gz -o aws-pcs-agent-v1.2.1-1.tar.gz
```

Você também pode obter a versão mais recente substituindo o número da versão pelo comando anterior (por exemplo:`aws-pcs-agent-v1-latest.tar.gz`). `latest`

> **Note**
>
> Isso pode mudar em futuras versões do software do agente AWS PCS.

5. (Opcional) Verifique a autenticidade e a integridade do pacote de software AWS PCS. Recomendamos que você faça isso para verificar a identidade do fornecedor do software e para verificar se a aplicação não foi alterada ou corrompida desde que foi publicada.

   a. Baixe a chave GPG pública para AWS PCS e importe-a para o seu chaveiro. *region*Substitua pelo Região da AWS local em que você iniciou sua instância temporária. O comando deve retornar um valor de chave. Registre o valor da chave; você o usa na próxima etapa.

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

   b. Execute o comando a seguir para verificar a impressão digital da chave GPG.

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

   O comando deve retornar uma impressão digital idêntica à seguinte:

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

   > **Important**
   >
   > Não execute o script de instalação do agente AWS PCS se a impressão digital não corresponder. Entrar em contato com o AWS Support.

c. Baixe o arquivo de assinatura e verifique a assinatura do arquivo tarball do software AWS PCS. *region*Substitua pelo Região da AWS local em que você iniciou sua instância temporária, como`us-east-1`.

```
wget https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.2.1-1.tar.gz.sig && \
    gpg --verify ./aws-pcs-agent-v1.2.1-1.tar.gz.sig
```

A saída deve ser semelhante ao seguinte:

```
gpg: assuming signed data in './aws-pcs-agent-v1.2.1-1.tar.gz'
gpg: Signature made Fri Dec 13 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
```

Se o resultado incluir `Good signature` e a impressão digital corresponder à impressão digital retornada na etapa anterior, vá para a próxima etapa.

> **Important**
>
> Não execute o script de instalação do software AWS PCS se a impressão digital não corresponder. Entrar em contato com o AWS Support.

6. Extraia os arquivos do `.tar.gz` arquivo compactado e navegue até o diretório extraído.

```
tar -xf aws-pcs-agent-v1.2.1-1.tar.gz && \
    cd aws-pcs-agent
```

7. Instale o software AWS PCS.

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

8. Verifique o arquivo da versão do software AWS PCS para confirmar uma instalação bem-sucedida.

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

A saída deve ser semelhante ao seguinte:

```
AGENT_INSTALL_DATE='Fri Dec 13 12:28:43 UTC 2024'
AGENT_VERSION='1.2.1'
AGENT_RELEASE='1'
```

Etapa 3 — Instalar o Slurm

Instale uma versão do Slurm compatível com AWS o PCS. Para obter mais informações, consulte Versões Slurm no PCS AWS.

> **Note**
>
> Se você tiver uma AMI com uma versão anterior do software Slurm instalada, deverá executar as etapas a seguir para instalar a nova versão do Slurm. O agente AWS PCS habilita a versão correta dos binários do Slurm em tempo de execução, de acordo com a versão do Slurm configurada no momento da criação do cluster.

Para instalar o Slurm

1. Conecte-se à mesma instância temporária em que você instalou o software AWS PCS.
2. Baixe o software instalador do Slurm. O instalador do Slurm é empacotado em um arquivo tarball () compactado. `.tar.gz` Para fazer download da última versão estável, use o seguinte comando: *`region`*Substitua pela Região da AWS da sua instância temporária, como`us-east-1`.

   ```
   curl https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.11-installer-24.11.5-1.tar.gz \
       -o aws-pcs-slurm-24.11-installer-24.11.5-1.tar.gz
   ```

   Você também pode obter a versão mais recente substituindo o número da versão pelo comando anterior (por exemplo:`aws-pcs-slurm-24.11-installer-latest.tar.gz`). `latest`

> **Note**
>
> Isso pode mudar em futuras versões do software instalador Slurm.

3. (Opcional) Verifique a autenticidade e a integridade do pacote de instalação do Slurm. Recomendamos que você faça isso para verificar a identidade do fornecedor do software e para verificar se a aplicação não foi alterada ou corrompida desde que foi publicada.

   a. Baixe a chave GPG pública para AWS PCS e importe-a para o seu chaveiro. *region*Substitua por Região da AWS onde você iniciou sua instância temporária. O comando deve retornar um valor de chave. Registre o valor da chave; você o usa na próxima etapa.

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

   b. Execute o comando a seguir para verificar a impressão digital da chave GPG.

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

   O comando deve retornar uma impressão digital idêntica à seguinte:

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

   > **Important**
   >
   > Não execute o script de instalação do Slurm se a impressão digital não corresponder. Entrar em contato com o AWS Support.

   c. Baixe o arquivo de assinatura e verifique a assinatura do arquivo tarball do instalador do Slurm. *region*Substitua pelo Região da AWS local em que você iniciou sua instância temporária, como`us-east-1`.

   ```
   wget https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.11-installer-24.11.5-1.tar.gz.sig && \
       gpg --verify ./aws-pcs-slurm-24.11-installer-24.11.5-1.tar.gz.sig
   ```

A saída deve ser semelhante ao seguinte:

```
gpg: assuming signed data in './aws-pcs-slurm-24.11-installer-24.11.5-1.tar.gz'
gpg: Signature made Wed May 14 14:23:38 2025 UTC
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
```

Se o resultado incluir `Good signature` e a impressão digital corresponder à impressão digital retornada na etapa anterior, vá para a próxima etapa.

> **Important**
>
> Não execute o script de instalação do Slurm se a impressão digital não corresponder. Entrar em contato com o AWS Support.

4. Extraia os arquivos do arquivo compactado `.tar.gz` e navegue para o diretório extraído.

```
tar -xf aws-pcs-slurm-24.11-installer-24.11.5-1.tar.gz && \
    cd aws-pcs-slurm-24.11-installer
```

5. Instale o Slurm. O instalador baixa, compila e instala o Slurm e suas dependências. Isso leva vários minutos, dependendo das especificações da instância temporária que você selecionou.

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

6. Verifique o arquivo da versão do agendador para confirmar a instalação.

```
cat /opt/aws/pcs/scheduler/slurm-24.11/version
```

A saída deve ser semelhante ao seguinte:

```
SLURM_INSTALL_DATE='Wed May 14 14:23:38 UTC 2025'
SLURM_VERSION='24.11.5'
PCS_SLURM_RELEASE='1'
```

Etapa 4 — (Opcional) Instale drivers, bibliotecas e software aplicativo adicionais

Instale drivers, bibliotecas e aplicativos adicionais na instância temporária. Os procedimentos de instalação variam de acordo com os aplicativos e bibliotecas específicos. Se você ainda não criou uma AMI personalizada para AWS PCS, recomendamos que primeiro crie e teste uma AMI com apenas o software AWS PCS e o Slurm instalados e, em seguida, adicione incrementalmente seu próprio software e configurações depois de confirmar o sucesso inicial.

Exemplos

- Software Elastic Fabric Adapter (EFA). Para obter mais informações, consulte Comece a usar o EFA e o MPI para cargas de trabalho de HPC na Amazon EC2 no Guia do usuário do Amazon Elastic Compute Cloud.
- Cliente Amazon Elastic File System (Amazon EFS). Para obter mais informações, consulte Instalação manual do cliente Amazon EFS no Guia do usuário do Amazon Elastic File System.
- Cliente Lustre, para usar o Amazon FSx for Lustre e o Amazon File Cache. Para obter mais informações, consulte Instalando o cliente Lustre no Guia do FSx usuário do Lustre.
- CloudWatch Agente da Amazon, para usar CloudWatch registros e métricas. Para obter mais informações, consulte Instalar o CloudWatch agente no Guia CloudWatch do usuário da Amazon.
- AWS Neuron, para usar os tipos de instância trn* e inf*. Para obter mais informações, consulte a documentação do AWS Neuron.
- Driver NVIDIA, CUDA e DCGM, para usar os tipos de instância p* ou g*.

Etapa 5 — Crie uma AMI compatível com AWS PCS

Depois de instalar os componentes de software necessários, você cria uma AMI que pode ser reutilizada para iniciar instâncias em grupos de nós de computação do AWS PCS.

Para criar uma AMI a partir de sua instância temporária

1. Abra o EC2 console da Amazon.
2. No painel de navegação, escolha Instâncias.
3. Selecione a instância temporária que você criou. Escolha Ações, Imagem, Criar imagem.
4. Em Create image (Criar imagem), faça o seguinte:

a. Em Image name (Nome da imagem), insira um nome descritivo para a AMI.

b. (Opcional) Em Image description (Descrição da imagem), informe a descrição do propósito da AMI.

c. Escolha Create Image (Criar imagem).

5. No painel de navegação, escolha AMIs.

6. Localize a AMI que você criou na lista. Aguarde até que seu status mude de Pendente para Disponível e use-o com um grupo de nós de computação AWS PCS.

Etapa 6 — Use a AMI personalizada com um grupo de nós de computação AWS PCS

Você pode usar sua AMI personalizada com um grupo de nós de computação AWS PCS novo ou existente.

New compute node group

Para usar a AMI personalizada

1. Abra o console AWS PCS.

2. No painel de navegação, escolha Clusters.

3. Escolha o cluster em que você usará a AMI personalizada e selecione grupos de nós de computação.

4. Crie um novo grupo de nós de computação. Para obter mais informações, consulte Criação de um grupo de nós de computação no AWS PCS. Em ID da AMI, pesquise o nome ou ID da AMI personalizada que você deseja usar. Conclua a configuração do grupo de nós de computação e escolha Criar grupo de nós de computação.

5. (Opcional) Confirme se a AMI oferece suporte a lançamentos de instâncias. Execute uma instância no grupo de nós de computação. Você pode fazer isso configurando o grupo de nós de computação para ter uma única instância estática ou enviar um trabalho para uma fila que usa o grupo de nós de computação.

   a. Verifique o EC2 console da Amazon até que uma instância apareça marcada com o novo ID do grupo de nós de computação. Para obter mais informações sobre isso, consulteEncontrando instâncias de grupos de nós de computação no AWS PCS..

b. Ao ver uma instância ser iniciada e concluir o processo de bootstrap, confirme se ela está usando a AMI esperada. Para fazer isso, selecione a instância e, em seguida, inspecione a ID da AMI em Detalhes. Ela deve corresponder à AMI que você configurou nas configurações do grupo de nós de computação.

c. (Opcional) Atualize a configuração de escalabilidade do grupo de nós de computação de acordo com seus valores preferidos.

Existing compute node group

Para usar a AMI personalizada

1. Abra o console AWS PCS.
2. No painel de navegação, escolha Clusters.
3. Escolha o cluster em que você usará a AMI personalizada e selecione grupos de nós de computação.
4. Selecione o grupo de nós que você deseja configurar e escolha Editar. Em ID da AMI, pesquise o nome ou ID da AMI personalizada que você deseja usar. Conclua a configuração do grupo de nós de computação e escolha Atualizar. As novas instâncias lançadas no grupo de nós de computação usarão a ID da AMI atualizada. As instâncias existentes continuarão usando a AMI antiga até que o AWS PCS as substitua. Para obter mais informações, consulte Atualização de um grupo de nós de computação AWS PCS.
5. (Opcional) Confirme se a AMI oferece suporte a lançamentos de instâncias. Execute uma instância no grupo de nós de computação. Você pode fazer isso configurando o grupo de nós de computação para ter uma única instância estática ou enviar um trabalho para uma fila que usa o grupo de nós de computação.

   a. Verifique o EC2 console da Amazon até que uma instância apareça marcada com o novo ID do grupo de nós de computação. Para obter mais informações sobre isso, consulteEncontrando instâncias de grupos de nós de computação no AWS PCS..

   b. Ao ver uma instância ser iniciada e concluir o processo de bootstrap, confirme se ela está usando a AMI esperada. Para fazer isso, selecione a instância e, em seguida, inspecione a ID da AMI em Detalhes. Ela deve corresponder à AMI que você configurou nas configurações do grupo de nós de computação.

   c. (Opcional) Atualize a configuração de escalabilidade do grupo de nós de computação de acordo com seus valores preferidos.

Etapa 7 — Encerrar a instância temporária

Depois de confirmar que sua AMI funciona conforme o esperado com o AWS PCS, você pode encerrar a instância temporária para parar de incorrer em cobranças por ela.

Para encerrar a instância temporária

1. Abra o EC2 console da Amazon.
2. No painel de navegação, escolha Instances (Instâncias).
3. Selecione a instância temporária que você criou e escolha Ações, Estado da instância, Encerrar instância.
4. Quando solicitado a confirmar, escolha Encerrar.

Instaladores de software para criar de forma personalizada AMIs para AWS PCS

AWS fornece um arquivo para download que pode instalar o software AWS PCS em uma instância. AWS também fornece software que pode baixar, compilar e instalar versões relevantes do Slurm e de suas dependências. Você pode usar essas instruções para criar uma versão personalizada AMIs para uso com o AWS PCS ou pode usar seus próprios métodos.

Sumário

AWS Instalador do software do agente PCS

O instalador do software do agente AWS PCS configura uma instância para funcionar com o AWS PCS durante o processo de inicialização da instância. Você deve usar AWS instaladores fornecidos para instalar o agente AWS PCS em sua AMI personalizada.

Para obter mais informações sobre o software do agente AWS PCS, consulteAWS Versões do agente PCS.

Instalador do Slurm

O instalador do Slurm baixa, compila e instala versões relevantes do Slurm e de suas dependências. Você pode usar o instalador do Slurm para criar de forma personalizada AMIs para AWS PCS. Você também pode usar seus próprios mecanismos se eles forem consistentes com a configuração de software fornecida pelo instalador do Slurm. Para obter mais informações sobre o suporte do AWS PCS para o Slurm, consulte. Versões Slurm no PCS AWS

O software AWS fornecido instala o seguinte:

- Slurm na versão principal e de manutenção solicitada (atualmente versão 24.11.x) - Licença GPL 2
  - O Slurm é construído com `--sysconfdir` um conjunto de `/etc/slurm`
  - O Slurm é construído com a opção e `--enable-pam` `--without-munge`
  - O Slurm é construído com a opção `--sharedstatedir=/run/slurm/`
  - O Slurm é construído com suporte a PMIX e JWT
  - O Slurm é instalado em `/opt/aws/pcs/schedulers/slurm-24.11`
- OpenPmix (versão 4.2.6) — Licença
  - O OpenPmix é instalado como um subdiretório do `/opt/aws/pcs/scheduler/`
- libjwt (versão 1.17.0) — Licença MPL-2.0
  - libjwt é instalado como um subdiretório do `/opt/aws/pcs/scheduler/`

O software AWS fornecido altera a configuração do sistema da seguinte forma:

- O `systemd` arquivo Slurm criado pela compilação é copiado `/etc/systemd/system/` com o nome do arquivo. `slurmd-24.11.service`
- Se eles não existirem, um usuário e um grupo (`slurm:slurm`) do Slurm são criados com UID/GID of. `401`
- No Amazon Linux 2 e no Rocky Linux 9, a instalação adiciona o repositório EPEL para instalar o software necessário para criar o Slurm ou suas dependências.
- RHEL9 Na instalação, habilitará `codeready-builder-for-rhel-9-rhui-rpms` e `epel-release-latest-9` instalará o software necessário `fedoraproject` para criar o Slurm ou suas dependências.

Sistemas operacionais compatíveis

Consulte Sistemas operacionais compatíveis no AWS PCS.

> **Note**
>
> AMIs de deep learning da AWS As versões (DLAMI) baseadas no Amazon Linux 2 e no Ubuntu 22.04 devem ser compatíveis com o software PCS e os instaladores AWS do Slurm. Para obter mais informações, consulte Escolhendo sua DLAMI no AMIs de deep learning da AWS Guia do desenvolvedor.

Tipos de instâncias compatíveis

AWS O software PCS e os instaladores do Slurm oferecem suporte a qualquer tipo de instância x86_64 ou arm64 que possa executar um dos sistemas operacionais compatíveis.

Versões do Slurm suportadas

Consulte Versões Slurm no PCS AWS.

Verifique os instaladores usando uma soma de verificação

Você pode usar SHA256 somas de verificação para verificar os arquivos tarball (.tar.gz) do instalador. Recomendamos que você faça isso para verificar a identidade do fornecedor do software e para verificar se a aplicação não foi alterada ou corrompida desde que foi publicada.

Para verificar um tarball

Use o utilitário sha256sum para a soma de SHA256 verificação e especifique o nome do arquivo tarball. Você deve executar o comando a partir do diretório em que salvou o arquivo tarball.

- SHA256

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

O comando deve retornar um valor de soma de verificação no formato a seguir.

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

Compare o valor da soma de verificação retornado pelo comando com o valor da soma de verificação fornecido na tabela a seguir. Se as somas de verificação corresponderem, é seguro executar o script de instalação.

> **Important**
>
> Se as somas de verificação não corresponderem, não execute o script de instalação. Entre em contato com a Suporte.

Por exemplo, o comando a seguir gera a SHA256 soma de verificação para o tarball do Slurm 24.11.5-1.

```
$ sha256sum aws-pcs-slurm-24.11-installer-24.11.5-1.tar.gz
```

Resultado do exemplo:

```
593efe4d66bef2f3e46d5a382fb5a32f7a3ca2510bcf1b3c85739f4f951810d5 aws-pcs-slurm-24.11-installer-24.11.5-1.tar.gz
```

As tabelas a seguir listam as somas de verificação das versões recentes dos instaladores. *us-east-1*Substitua pelo Região da AWS local em que você usa o AWS PCS.

AWS Agente PCS

Installer (Instalador)	Faça download do URL	SHA256 soma de verificação
AWS Agente PCS 1.2.1-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.2.1-1.tar.gz`	`2b784643ca01ccca1baa64fbfb34bb41efe8bdca69470998b74ce3962bc271d4`
AWS Agente PCS 1.2.0-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-agent/aws-pcs`	`470db8c4fc9e50277b6317f98584b6b547e73523043e34f018eecae767846805`

Installer (Instalador)	Faça download do URL	SHA256 soma de verificação
	`-agent-v1.2.0-1.ta r.gz`	
AWS Agente PCS 1.1.1-1	`https://aws-pcs-re po- us-east-1 .s3.us- east-1 .amazonaws.com/ aws-pcs-agent/aws-pcs -agent-v1.1.1-1.ta r.gz`	`bef078bf60a6d8ecde 2e6c49cd34d088703f 02550279e3bf483d57 a235334dc6`
AWS Agente PCS 1.1.0-1	`https://aws-pcs-re po- us-east-1 .s3.us- east-1 .amazonaws.com/ aws-pcs-agent/aws-pcs -agent-v1.1.0-1.ta r.gz`	`594c32194c71bccc5d 66e5213213ae38dd2c 6d2f9a950bb01accea 0bbab0873a`
AWS Agente PCS 1.0.1-1	`https://aws-pcs-re po- us-east-1 .s3.us- east-1 .amazonaws.com/ aws-pcs-agent/aws-pcs -agent-v1.0.1-1.ta r.gz`	`04e22264019837e3f4 2d8346daf5886eaace cd21571742eb505ea8 911786bcb2`
AWS Agente PCS 1.0.0-1	`https://aws-pcs-re po- us-east-1 .s3.us- east-1 .amazonaws.com/ aws-pcs-agent/aws-pcs -agent-v1.0.0-1.ta r.gz`	`d2d3d68d00c685435c 38af471d7e2492dde5 ce9eb222d7b6ef0042 144b134ce0`

Instalador do Slurm

Installer (Instalador)	Faça download do URL	SHA256 soma de verificação
Slurm 24.11.5-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.11-installer-24.11.5-1.tar.gz`	`593efe4d66bef2f3e46d5a382fb5a32f7a3ca2510bcf1b3c85739f4f951810d5`
Slurm 24.05.7-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.05-installer-24.05.7-1.tar.gz`	`0b5ed7c81195de2628c78f37c79e63fc4ae99132ca6b019b53a0d68792ee82c5`
Slurm 24.05.5-2	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.05-installer-24.05.5-2.tar.gz`	`7cc8d8294f2fbff95fe0602cf9e21e02003b5d96c0730e0a18c6aa04c7a4967b`
Slurm 23.11.10-3	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.10-3.tar.gz`	`488a10ee0fbd57ec0e0ff7ea708a9e3038fafdc025c6bb391c75c2e2a7852a00`
Slurm 23.11.10-2	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-insta`	`0bbe85423305c05987931168caf98da08a34c25f9eec0690e8e74de0b7bc8752`

Installer (Instalador)	Faça download do URL	SHA256 soma de verificação
	`ller-23.11.10-2.ta r.gz`	
Slurm 23.11.10-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.10-1.tar.gz`	`27e8faa9980e92cdfd8cfdc71f937777f0934552ce61e33dac4ecf5a20321e44`
Slurm 23.11.9-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz`	`1de7d919c8632fe8e2806611bed4fde1005a4fadc795412456e935c7bba2a9b8`

Notas de lançamento da amostra AWS PCS AMIs

AMIs para obter as versões principais suportadas mais recentes do agendador, receba atualizações de segurança e correções críticas de bugs. Esses patches de segurança incrementais não estão incluídos nas notas oficiais de lançamento.

> **Important**
>
> Amostras AMIs relacionadas às versões antigas do agendador não são suportadas e não recebem atualizações.

> **Important**
>
> AMIs As amostras são para fins de demonstração e não são recomendadas para cargas de trabalho de produção.

Sumário

AWS Exemplo de PCS AMIs para x86_64 (Amazon Linux 2)

Fauna 24.11

> **Note**
>
> AWS O PCS suporta a contabilização do Slurm 24.11 e versões posteriores. Para obter mais informações, consulte Contabilidade de slurm no PCS AWS.

Nome da AMI

- `aws-pcs-sample_ami-amzn2-x86_64-slurm-24.11`

EC2 Instâncias suportadas

- Todas as instâncias com um processador x86 de 64 bits. Para encontrar instâncias compatíveis, navegue até o EC2 console da Amazon. Escolha Tipos de instância e, em seguida, pesquise por`Architectures=x86_64`.

Conteúdo da AMI

- AWS Serviço suportado: AWS PCS
- Sistema operacional: Amazon Linux 2
- Arquitetura de computação: x86_64
- Tipo de volume do EBS: gp2
- Instalador EFA: 1.33.0
- GDRCopy: 2,4
- Driver NVIDIA: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Fauna 24.05

Nome da AMI

- `aws-pcs-sample_ami-amzn2-x86_64-slurm-24.05`

EC2 Instâncias suportadas

- Todas as instâncias com um processador x86 de 64 bits. Para encontrar instâncias compatíveis, navegue até o EC2 console da Amazon. Escolha Tipos de instância e, em seguida, pesquise por`Architectures=x86_64`.

Conteúdo da AMI

- AWS Serviço suportado: AWS PCS
- Sistema operacional: Amazon Linux 2
- Arquitetura de computação: x86_64
- Tipo de volume do EBS: gp2
- Instalador EFA: 1.33.0
- GDRCopy: 2,4
- Driver NVIDIA: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Fauna 23.11

Nome da AMI

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

EC2 Instâncias suportadas

- Todas as instâncias com um processador x86 de 64 bits. Para encontrar instâncias compatíveis, navegue até o EC2 console da Amazon. Escolha Tipos de instância e, em seguida, pesquise por`Architectures=x86_64`.

Conteúdo da AMI

- AWS Serviço suportado: AWS PCS
- Sistema operacional: Amazon Linux 2
- Arquitetura de computação: x86_64
- Tipo de volume do EBS: gp2
- Instalador EFA: 1.33.0
- GDRCopy: 2,4
- Driver NVIDIA: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

AWS Amostra de PCS AMIs para Arm64 (Amazon Linux 2)

Fauna 24.11

> **Note**
>
> AWS O PCS suporta a contabilização do Slurm 24.11 e versões posteriores. Para obter mais informações, consulte Contabilidade de slurm no PCS AWS.

Nome da AMI

- `aws-pcs-sample_ami-amzn2-arm64-slurm-24.11`

EC2 Instâncias suportadas

- Todas as instâncias com um processador Arm de 64 bits. Para encontrar instâncias compatíveis, navegue até o EC2 console da Amazon. Escolha Tipos de instância e, em seguida, pesquise por`Architectures=arm64`.

Conteúdo da AMI

- AWS Serviço suportado: AWS PCS
- Sistema operacional: Amazon Linux 2

- Arquitetura de computação: arm64
- Tipo de volume do EBS: gp2
- Instalador EFA: 1.33.0
- GDRCopy: 2,4
- Driver NVIDIA: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Fauna 24.05

Nome da AMI

- `aws-pcs-sample_ami-amzn2-arm64-slurm-24.05`

EC2 Instâncias suportadas

- Todas as instâncias com um processador Arm de 64 bits. Para encontrar instâncias compatíveis, navegue até o EC2 console da Amazon. Escolha Tipos de instância e, em seguida, pesquise por`Architectures=arm64`.

Conteúdo da AMI

- AWS Serviço suportado: AWS PCS
- Sistema operacional: Amazon Linux 2
- Arquitetura de computação: arm64
- Tipo de volume do EBS: gp2
- Instalador EFA: 1.33.0
- GDRCopy: 2,4
- Driver NVIDIA: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Fauna 23.11

Nome da AMI

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

EC2 Instâncias suportadas

- Todas as instâncias com um processador Arm de 64 bits. Para encontrar instâncias compatíveis, navegue até o EC2 console da Amazon. Escolha Tipos de instância e, em seguida, pesquise por`Architectures=arm64`.

Conteúdo da AMI

- AWS Serviço suportado: AWS PCS
- Sistema operacional: Amazon Linux 2
- Arquitetura de computação: arm64
- Tipo de volume do EBS: gp2
- Instalador EFA: 1.33.0
- GDRCopy: 2,4
- Driver NVIDIA: 550.127.08
- NVIDIA CUDA: 12.4.1_550.54.15

Sistemas operacionais compatíveis no AWS PCS

AWS O PCS usa a Amazon Machine Image (AMI) configurada para um grupo de nós de computação para iniciar EC2 instâncias nesse grupo de nós de computação. A AMI determina o sistema operacional que as EC2 instâncias usam. Você não pode alterar o sistema operacional na amostra AWS PCS AMIs. Você deve criar uma AMI personalizada se quiser usar um sistema operacional diferente. Para obter mais informações, consulte Amazon Machine Images (AMIs) para AWS PCS.

Sistemas operacionais compatíveis

- Amazon Linux 2

  Esse é o sistema operacional na amostra AWS PCS AMIs.

  > **Important**
  >
  > AMIs As amostras são para fins de demonstração e não são recomendadas para cargas de trabalho de produção. Você deve criar e usar uma AMI personalizada para cargas de trabalho de produção, mesmo que pretenda usar o Amazon Linux 2.

- RedHat Linux corporativo 9 (RHEL 9)

  O custo sob demanda do RHEL de qualquer tipo de instância é maior do que para outros sistemas operacionais compatíveis. Para ter mais informações sobre preços, consulte Definição de preço sob demanda e How is Red Hat Enterprise Linux on Amazon Elastic Compute Cloud offered and priced?.

- Rocky Linux 9

  Você pode usar o Rocky Linux 9 oficial AMIs como base para uma AMI personalizada. Sua compilação personalizada da AMI pode falhar se a AMI básica não tiver o kernel mais recente.

  Para atualizar o kernel

  1. Execute uma instância usando um ID de AMI rocky9 aqui: https://rockylinux.org/cloud-images/
  2. ssh na instância e execute o seguinte comando:

     ```
     sudo yum -y update
     ```

3. Crie uma imagem da instância. Você especifica essa imagem como a da `ParentImage` sua AMI personalizada.

- Ubuntu 22.04

  O Ubuntu 22.04 requer chaves mais seguras para SSH e não suporta chaves RSA por padrão. Recomendamos que você gere e use uma ED25519 chave em vez disso.

AWS Versões do agente PCS

O software do agente AWS PCS configura as EC2 instâncias que o AWS PCS executa para uso com o Slurm. Você inclui o agente em uma Amazon Machine Images (AMI) que você especifica ao criar grupos de nós de computação para seu cluster. As EC2 instâncias executadas nesses grupos de nós de computação usam a AMI especificada e o software agente AWS PCS incluído. O agente AWS PCS permite que uma EC2 instância se registre como parte do cluster. Para usar o software de agente AWS PCS mais recente, você deve atualizar seu software personalizado AMIs. Para obter mais informações, consulte Etapa 2 — Instalar o agente AWS PCS em Imagens personalizadas da Amazon Machine (AMIs) para AWS PCS.

AWS Versão do agente PCS	Data de lançamento	Notas de lançamento
v1.2.0-1	7 de março de 2025	• Suporte habilitado para IPv6 entradas `slurmd.conf` .
v1.1.1-1	13 de dezembro de 2024	• Corrigido um problema em que uma versão incorreta do Slurm era relatada na chamada para. RegisterComputeNodeGroupInstance • Corrigido um problema em que os metadados da instância não eram buscados corretamente se um script personalizado `/opt/aws/pcs/etc/bootstrap_hooks/` fosse executado.
v1.1.0-1	06 de dezembro de 2024	• Habilitou `/opt/aws/pcs/etc/bootstrap_hooks/` a execução de scripts personalizados antes das etapas de bootstrap.

AWS Versão do agente PCS	Data de lançamento	Notas de lançamento
v1.0.1-1	22 de outubro de 2024	• Corrigido um problema em que os dispositivos NVIDIA não funcionavam quando `slurmd` iniciados em instâncias habilitadas para GPU.
v1.0.0-1	28 de agosto de 2024	• Versão inicial.

Versões Slurm no PCS AWS

O SchedMD aprimora continuamente o Slurm com novos recursos, otimizações e patches de segurança. O SchedMD lança uma nova versão principal em intervalos regulares e planeja oferecer suporte a até 3 versões a qualquer momento. AWS O PCS foi projetado para atualizar automaticamente o controlador Slurm com versões de patch.

Quando o SchedMD encerra o suporte para uma versão principal específica, o AWS PCS também encerra o suporte para essa versão principal. AWS O PCS envia um aviso prévio se uma versão principal do Slurm estiver próxima do fim de sua vida útil, para ajudar os clientes a saberem quando atualizar seus clusters para uma versão mais recente compatível.

Recomendamos que você use a versão mais recente compatível do Slurm para implantar seu cluster e acessar os avanços e melhorias mais recentes.

Versões do Slurm suportadas no PCS AWS

A tabela a seguir mostra as versões suportadas do Slurm e as datas e informações importantes de cada versão.

Versão Slurm	Data de lançamento do SchedMD	AWS Data de lançamento do PCS	Data de fim do suporte ao AWS PCS	Versão mínima compatível do agente AWS PCS	Amostra de AWS PCS compatível AMIs
24.11	29/11/2024	14/05/2025	31/05/2026	1.0.0-1	• `aws-pcs-sample_ami-amzn2-x86_64-slurm-24.11` • `aws-pcs-sample_ami`

Versão Slurm	Data de lançamento do SchedMD	AWS Data de lançamento do PCS	Data de fim do suporte ao AWS PCS	Versão mínima compatível do agente AWS PCS	Amostra de AWS PCS compatível AMIs
					`-amzn2-arm64-slurm-24.11`
24.05	30/05/2024	18/12/2024	30/11/2025	1.0.0-1	• `aws-pcs-sample_ami-amzn2-x86_64-slurm-24.05` • `aws-pcs-sample_ami-amzn2-arm64-slurm-24.05`

Versões do Slurm não suportadas no PCS AWS

A tabela a seguir mostra as versões do Slurm que não são suportadas no AWS PCS.

Versão Slurm	Data de lançamento do SchedMD	AWS Data de lançamento do PCS	Data de fim do suporte ao AWS PCS		
23.11	21/11/2023	28/08/2024	31/05/2025		

Notas de lançamento das versões do Slurm no PCS AWS

Este tópico descreve mudanças importantes para cada versão do Slurm atualmente suportada no AWS PCS. Recomendamos que você analise as alterações entre a versão antiga e a nova ao atualizar seu cluster.

Fauna 24.11

Mudanças implementadas no AWS PCS

- AWS O PCS oferece suporte à contabilidade do Slurm. Para obter mais informações, consulte Contabilidade de slurm no PCS AWS.

Para obter mais informações sobre o Slurm 24.11, consulte as seguintes publicações:

- Anúncio de lançamento do SchedMD
- Notas de lançamento do SchedMD

Fauna 24.05

Mudanças implementadas no AWS PCS

- O novo módulo Slurm Step Manager agora está habilitado por padrão no AWS PCS. Esse módulo oferece benefícios significativos ao transferir o gerenciamento de etapas do controlador central para os nós de computação, melhorando substancialmente a simultaneidade do sistema em ambientes com uso intenso de etapas. Para suportar essa configuração e melhor isolar `Prolog` e `Epilog` processar a execução, novos sinalizadores de prólogo (`Contain`,`Alloc`) são habilitados.
- A comunicação hierárquica do controlador para os nós de computação é habilitada para otimizar a comunicação entre nós do Slurm, o que melhora a escalabilidade e o desempenho. Além disso, a configuração de roteamento agora usa listas de nós de partição para comunicações do controlador, em vez do algoritmo de roteamento padrão do plug-in, aprimorando a resiliência do sistema.
- Um novo plugin de hash `HashPlugin=hash/sha3` substitui o anterior. `hash/k12 plugin` Agora, isso está habilitado por padrão nos clusters AWS PCS.
- Os registros do controlador Slurm agora incluem recursos aprimorados de auditoria para todas as chamadas de procedimento remoto (RPC) de entrada para. `slurmctld` Os registros incluem o endereço de origem, o usuário autenticado e o tipo de RPC antes do processamento da conexão.

Para obter mais informações sobre o Slurm 24.05, consulte as seguintes publicações:

- Anúncio de lançamento do SchedMD
- Notas de lançamento do SchedMD

Fauna 23.11

Configurações do Slurm que você pode alterar no PCS AWS

- O `SuspendTime` padrão é. `60` Use o parâmetro `scaleDownIdleTimeInSeconds` de configuração AWS PCS para defini-lo. Para obter mais informações, consulte o `scaleDownIdleTimeInSeconds`parâmetro do tipo de `ClusterSlurmConfiguration` dados na Referência da API AWS PCS.
- O `MaxJobCount` e `MaxArraySize` é baseado no tamanho que você escolher para o cluster. Para obter mais informações, consulte o `size`parâmetro da ação da `CreateCluster` API na Referência da API AWS PCS.
- A configuração do `SelectTypeParameters` Slurm é padronizada como. `CR_CPU` Você pode fornecê-lo como um valor `slurmCustomSettings` para defini-lo ao criar um cluster. Para obter mais informações, consulte o `slurmCustomSettings`parâmetro da ação da `CreateCluster` API e SlurmCustomSettingna Referência da API AWS PCS.
- Você pode definir `Prolog` e `Epilog` no nível do cluster. Você pode fornecê-lo como um valor `slurmCustomSettings` para defini-lo ao criar um cluster. Para obter mais informações, consulte `CreateCluster`e SlurmCustomSettingna Referência da API AWS PCS.
- Você pode definir `Weight` e `RealMemory` no nível do grupo de nós de computação. Você pode fornecê-lo como um valor `slurmCustomSettings` para defini-lo ao criar um grupo de nós de computação. Para obter mais informações, consulte `CreateComputeNodeGroup`e SlurmCustomSettingna Referência da API AWS PCS.

Perguntas frequentes sobre as versões do Slurm no PCS AWS

AWS O PCS mantém o suporte para várias versões do Slurm. Quando uma nova versão do Slurm é introduzida, o AWS PCS fornece suporte técnico e patches de segurança até que essa versão chegue ao fim do suporte (EOS) do SchedMD. AWS PCS se refere à data EOS de uma versão do Slurm como fim da vida útil (EOL) para ser consistente com a terminologia. AWS

Por quanto tempo o AWS PCS suporta a versão Slurm?

AWS O suporte do PCS para as versões do Slurm está alinhado com os ciclos de suporte do SchedMD para as versões principais. AWS O PCS suporta a versão atual e as duas versões principais anteriores mais recentes. Quando o SchedMD lança uma nova versão principal, o AWS PCS encerra o suporte para a versão mais antiga suportada. AWS O PCS lança novas versões principais do Slurm o mais rápido possível, mas pode haver um atraso entre o lançamento do SchedMD e sua disponibilidade no PCS. AWS

Como meus clusters obtêm novos lançamentos da versão de patch do Slurm?

Para resolver bugs e correções de segurança, o AWS PCS foi projetado para aplicar automaticamente patches aos controladores de cluster que são executados em contas internas de propriedade do serviço. Para instalar patches em EC2 instâncias em sua Conta da AWS, atualize a Amazon Machine Image (AMI) para seus grupos de nós de computação e atualize os grupos de nós de computação para usar a AMI atualizada. Para obter mais informações, consulte Imagens personalizadas da Amazon Machine (AMIs) para AWS PCS.

> **Note**
>
> Os controladores Slurm não estão disponíveis enquanto os atualizamos. Os trabalhos em execução não são afetados. Os trabalhos enviados antes que o controlador do cluster fique indisponível são retidos até que o controlador esteja disponível.

Como sou informado sobre um próximo evento de EOL da versão Slurm?

Enviamos uma mensagem de e-mail 6 meses antes da data de EOL. Enviamos uma mensagem de e-mail a cada mês antes do EOL, com uma mensagem de e-mail final 1 semana antes da data do EOL. Após a data de EOL, enviamos mensagens de e-mail mensais por 12 meses para clientes que executam clusters AWS PCS com versões do EOL Slurm. Podemos suspender um cluster com uma versão do EOL Slurm se forem identificadas vulnerabilidades de segurança para essa versão.

Como posso determinar se a versão do Slurm usada pelo meu cluster está executando uma versão do EOL Slurm?

Enviamos uma mensagem de e-mail para notificá-lo de que você tem um cluster em execução com uma versão do EOL Slurm. Publicamos um alerta nos AWS Health Dashboard alertas que contém os detalhes de seus clusters com as versões do EOL Slurm. Você também pode usar o console AWS PCS para identificar os clusters com versões do EOL Slurm.

O que devo fazer se minha versão do Slurm estiver próxima ou além do EOL?

Crie um novo cluster com uma versão mais recente compatível do Slurm e atualize a versão do Slurm em seu grupo de nós de computação. AMIs A versão do Slurm nas suas EC2 instâncias AMIs e em execução não pode estar mais do que duas versões atrás da versão do Slurm do cluster. Para obter mais informações, consulte Imagens personalizadas da Amazon Machine (AMIs) para AWS PCS.

O que acontecerá se eu não mudar para uma versão mais recente do Slurm até a data de EOL?

Você não pode criar novos clusters com uma versão do EOL Slurm. Os clusters existentes podem operar por até 12 meses sem AWS suporte, e nenhuma ação imediata é necessária para manter sua operação. Após a data de EOL, o suporte, as atualizações de segurança e a disponibilidade não são garantidos. Podemos suspender um cluster por motivos de segurança. É altamente recomendável que você use uma versão compatível do Slurm para manter a segurança e o suporte de seus clusters AWS PCS.

Quais são os riscos de operar um cluster com versões do EOL Slurm?

Clusters com versões do EOL Slurm apresentam riscos operacionais e de segurança significativos. Sem o monitoramento ativo do SchedMD, as vulnerabilidades de segurança podem permanecer sem serem detectadas ou resolvidas. Se vulnerabilidades críticas forem descobertas, poderemos suspender seus clusters imediatamente.

O que acontece com meus trabalhos, recursos de computação, armazenamento e rede do cluster quando meu cluster é suspenso?

Todos os recursos gerenciados pelo AWS PCS são encerrados. Isso inclui o controlador Slurm, grupos de nós de computação e instâncias. EC2 Todos os trabalhos executados em instâncias de computação são encerrados imediatamente e o cluster entra em um estado suspenso. Os recursos gerenciados pelo cliente, como sistemas de arquivos externos, permanecem intactos. Você pode usar o console AWS PCS e as ações da API para acessar a configuração do cluster.

Posso reiniciar um cluster suspenso para retomar seus trabalhos restantes?

Não, você não pode reiniciar um cluster suspenso. Você pode usar a configuração do cluster suspenso para criar um novo cluster com uma versão compatível do Slurm. Você pode executar os trabalhos restantes se os salvou em um sistema de arquivos externo.

Posso solicitar uma extensão além do período de carência de 12 meses?

Não, você não pode solicitar uma extensão para executar seu cluster além do período de carência de 12 meses. Oferecemos um prazo estendido para ajudá-lo a mudar para uma versão compatível do

Slurm. Para evitar interrupções nas operações do cluster, recomendamos que você alterne antes que sua versão do Slurm atinja o EOL.

Contabilidade de slurm no PCS AWS

Você pode habilitar a contabilização em seus novos clusters AWS PCS para monitorar o uso do cluster, impor limites de recursos e gerenciar um controle de acesso refinado a filas específicas ou grupos de nós de computação. AWS O PCS cria e gerencia o banco de dados contábil do seu cluster, eliminando a necessidade de criar e gerenciar seu próprio banco de dados contábil separado. AWS O PCS usa o recurso de contabilidade no Slurm. Para obter mais informações sobre o recurso de contabilidade no Slurm, consulte a documentação do Slurm em SchedMD.

Para usar a contabilidade, ative-a ao criar um novo cluster e, opcionalmente, definir parâmetros contábeis. Depois que o status do cluster for `Active` e tiver grupos de nós de computação, você poderá se conectar ao shell Linux de um nó de login para realizar funções contábeis, como visualizar dados do trabalho com o comando Slurm`sacct`.

> **Note**
>
> A contabilidade é compatível com o Slurm 24.11 ou posterior.

AWS PCS console

Na página Criar cluster, você deve selecionar uma versão válida do Slurm (versão 24.11 ou posterior). Em Configurações do Agendador, habilite Contabilidade.

AWS PCS API

Forneça a `accounting` configuração em sua chamada para a ação `CreateCluster` da API. No `accounting` objeto, defina o `mode` para`STANDARD`. Para obter mais informações, consulte CreateClusterContabilidade na Referência da API AWS PCS.

O exemplo a seguir usa o AWS CLI para chamar a ação `CreateCluster` da API. A substring do valor do parâmetro permite a `accounting='{mode=STANDARD}'` contabilização.

```
aws pcs create-cluster --cluster-name cluster-name \
                       --scheduler type=SLURM,version=24.11 \
                       --size SMALL \
                       --networking subnetIds=cluster-subnet-
id,securityGroupIds=cluster-security-group-id \
                       --slurm-configuration
 scaleDownIdleTimeInSeconds=180,accounting='{mode=STANDARD}',slurmCustomSettings='[{paramete
```

> **Important**
>
> Você receberá cobranças de cobrança adicionais se ativar a contabilidade. Para obter mais informações, consulte a página de preços do AWS PCS.

> **Important**
>
> Você não pode desativar a contabilização em um cluster que a tenha habilitada. Você deve excluir o cluster.

Conceitos-chave para contabilidade Slurm no PCS AWS

Os conceitos a seguir são específicos do AWS PCS e controlam como AWS o PCS implementa a contabilidade do Slurm.

Banco de dados de contabilidade

AWS O PCS armazena seus dados contábeis em um banco de dados criado em um banco de dados Conta da AWS que AWS possui. Você não tem acesso ao `slurmdbd.conf`.

Tempo de purga padrão

Essa configuração de AWS PCS especifica o período de retenção (em dias) para todos os tipos de registros contábeis (trabalhos, eventos, reservas, etapas, suspensões, transações, dados de uso). Por exemplo, se o valor for 30, o AWS PCS retém os registros contábeis por 30 dias. Você fornece esse valor ao criar o cluster. Se você não fornecer um valor, o AWS PCS reterá os registros contábeis no banco de dados indefinidamente.

AWS PCS console

Você especifica o tempo de limpeza padrão como parte das etapas para criar um cluster. Na página Criar cluster, você deve selecionar uma versão válida do Slurm (versão 24.11 ou posterior) e ativar a contabilização. Em Configurações do Agendador, forneça um valor inteiro para o tempo de limpeza padrão (dias).

AWS PCS API

Especifique o `defaultPurgeTimeInDays` como parte das `accounting` informações que você fornece em sua chamada para a ação da `CreateCluster` API. Para obter mais informações, consulte CreateClusterContabilidade na Referência da API AWS PCS.

> **Note**
>
> Quando você usa a API AWS PCS para criar um cluster, o valor padrão para `defaultPurgeTimeInDays` é `-1` e `0` não é um valor válido.

Aplicação da política contábil

Essa configuração determina com que rigor o Slurm aplica as regras de envio de trabalhos, os limites de recursos e as políticas contábeis para seu cluster. Essa configuração corresponde ao `AccountingStorageEnforce` parâmetro no `slurm.conf` arquivo do seu cluster. Você pode selecionar qualquer combinação de opções de fiscalização. Se você não selecionar nenhuma opção, não haverá restrições contábeis aplicadas aos trabalhos no cluster. AWS O PCS suporta as seguintes opções:

- associações — job-to-account mapeamento
- limites — restrições de recursos
- QoS — requisitos de qualidade de serviço
- modo de segurança — conclusão garantida dentro dos limites
- nosteps — desativa a contabilização de etapas
- nojobs — desativa a contabilização de tarefas

Para obter mais informações sobre essas opções, consulte a documentação do Slurm em SchedMD.

AWS PCS console

Você define as opções como parte das etapas para criar um cluster. Na página Criar cluster, você deve selecionar uma versão válida do Slurm (versão 24.11 ou posterior) e ativar a contabilização. Selecione as opções desejadas na lista suspensa Aplicação da política contábil em Configurações do Agendador.

AWS PCS API

No Slurm, essas opções são definidas no arquivo de um cluster. `slurm.conf` Você não tem acesso direto ao `slurm.conf` para seu cluster AWS PCS. Em vez disso, você fornece `SlurmCustomSettings` à `CreateCluster` API a ação ao criar um cluster. Para obter mais informações, consulte CreateClustera Referência da API AWS PCS.

Obtenha a configuração contábil para um cluster AWS PCS existente

A configuração de contabilidade do Slurm está incluída na configuração do Slurm do seu cluster.

AWS PCS console

1. Escolha Clusters no painel de navegação.
2. Escolha o nome do cluster na lista.
3. Na guia Configuração, encontre a configuração contábil em Configuração do Slurm

AWS PCS API

Use a ação `GetCluster` da API para obter a configuração do cluster. Você pode encontrar a configuração contábil no`slurmConfiguration`. A configuração para `mode` e o valor de `defaultPurgeTimeInDays` estão abaixo`accounting`. As opções selecionadas de aplicação da política contábil estão em`slurmCustomSettings`. Para obter mais informações, consulte GetClustera Referência da API AWS PCS.

Segurança no serviço de computação AWS paralela

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de data centers e arquiteturas de rede criados para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O modelo de responsabilidade compartilhada descreve isso como segurança da nuvem e segurança na nuvem:

- Segurança da nuvem — AWS é responsável por proteger a infraestrutura que executa AWS os serviços no Nuvem AWS. AWS também fornece serviços que você pode usar com segurança. Auditores terceirizados testam e verificam regularmente a eficácia de nossa segurança como parte dos Programas de Conformidade Programas de AWS de . Para saber mais sobre os programas de conformidade que se aplicam ao Serviço de Computação AWS Paralela, consulte AWS Serviços no escopo do programa de conformidade AWS .
- Segurança na nuvem — Sua responsabilidade é determinada pelo AWS serviço que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da empresa e as leis e regulamentos aplicáveis.

Esta documentação ajuda você a entender como aplicar o modelo de responsabilidade compartilhada ao usar o AWS PCS. Os tópicos a seguir mostram como configurar o AWS PCS para atender aos seus objetivos de segurança e conformidade. Você também aprenderá a usar outros AWS serviços que ajudam a monitorar e proteger seus recursos de AWS PCS.

Tópicos

Proteção de dados no serviço de computação AWS paralela

O modelo de responsabilidade AWS compartilhada O modelo se aplica à proteção de dados no Serviço de Computação AWS Paralela. Conforme descrito neste modelo, AWS é responsável por proteger a infraestrutura global que executa todos os Nuvem AWS. Você é responsável por manter o controle sobre o conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos Serviços da AWS que usa. Para obter mais informações sobre a privacidade de dados, consulte as Data Privacy FAQ. Para obter mais informações sobre a proteção de dados na Europa, consulte a postagem do blog AWS Shared Responsibility Model and RGPD no Blog de segurança da AWS .

Para fins de proteção de dados, recomendamos que você proteja Conta da AWS as credenciais e configure usuários individuais com AWS IAM Identity Center ou AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use uma autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com AWS os recursos. Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Configure a API e o registro de atividades do usuário com AWS CloudTrail. Para obter informações sobre o uso de CloudTrail trilhas para capturar AWS atividades, consulte Como trabalhar com CloudTrail trilhas no Guia AWS CloudTrail do usuário.
- Use soluções de AWS criptografia, juntamente com todos os controles de segurança padrão Serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sigilosos armazenados no Amazon S3.
- Se você precisar de módulos criptográficos validados pelo FIPS 140-3 ao acessar AWS por meio de uma interface de linha de comando ou de uma API, use um endpoint FIPS. Para obter mais informações sobre os endpoints FIPS disponíveis, consulte Federal Information Processing Standard (FIPS) 140-3.

É altamente recomendável que nunca sejam colocadas informações confidenciais ou sigilosas, como endereços de e-mail de clientes, em tags ou campos de formato livre, como um campo Nome. Isso inclui quando você trabalha com o AWS PCS ou outro Serviços da AWS usando o console AWS CLI, a API ou AWS SDKs. Quaisquer dados inseridos em tags ou em campos de texto de formato livre

usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se você fornecer um URL para um servidor externo, recomendamos fortemente que não sejam incluídas informações de credenciais no URL para validar a solicitação a esse servidor.

Criptografia em repouso

A criptografia é ativada por padrão para dados em repouso quando você cria um cluster de Serviço de Computação AWS Paralela (AWS PCS) com a AWS Management Console AWS CLI,, API AWS PCS ou AWS SDKs. AWS O PCS usa uma AWS chave KMS própria para criptografar dados em repouso. Para obter mais informações, consulte Chaves do cliente e AWS chaves no Guia do AWS KMS desenvolvedor. Você também pode usar uma chave gerenciada pelo cliente. Para obter mais informações, consulte Política de chave KMS necessária para uso com volumes criptografados do EBS no PCS AWS.

O segredo do cluster é armazenado AWS Secrets Manager e criptografado com a chave KMS gerenciada pelo Secrets Manager. Para obter mais informações, consulte Trabalhando com segredos de cluster no AWS PCS.

Em um cluster AWS PCS, os seguintes dados estão em repouso:

- Estado do agendador — inclui dados sobre trabalhos em execução e nós provisionados no cluster. Esses são os dados nos quais o Slurm persiste, `StateSaveLocation` conforme definido em seu. `slurm.conf` Para obter mais informações, consulte a descrição StateSaveLocationna documentação do Slurm. AWS O PCS exclui os dados do trabalho após a conclusão de um trabalho.
- Segredo de autenticação do agendador — O AWS PCS o usa para autenticar todas as comunicações do agendador no cluster.

Para obter informações sobre o estado do agendador, o AWS PCS criptografa automaticamente os dados e os metadados antes de gravá-los no sistema de arquivos. O sistema de arquivos criptografados usa o algoritmo de criptografia AES-256 padrão do setor para dados em repouso.

Criptografia em trânsito

Suas conexões com a API AWS PCS usam criptografia TLS com o processo de assinatura Signature Version 4, independentemente de você usar o AWS Command Line Interface (AWS CLI) ou AWS SDKs. Para obter mais informações, consulte Assinatura de solicitações de AWS API no Guia AWS

Identity and Access Management do usuário. AWS gerencia o controle de acesso por meio da API com as políticas do IAM para as credenciais de segurança que você usa para se conectar.

AWS O PCS usa o TLS para se conectar a outros AWS serviços.

Em um cluster do Slurm, o agendador é configurado com o plug-in de autenticação que fornece `auth/slurm` autenticação para todas as comunicações do agendador. O Slurm não fornece criptografia no nível do aplicativo para suas comunicações. Todos os dados que fluem pelas instâncias do cluster permanecem locais na VPC e, portanto, estão sujeitos à criptografia da EC2 VPC se essas instâncias oferecerem suporte à criptografia em trânsito. Para obter mais informações, consulte Criptografia em trânsito no Guia do usuário do Amazon Elastic Compute Cloud. A comunicação é criptografada entre o controlador (provisionado em uma conta de serviço) e os nós do cluster em sua conta.

Gerenciamento de chaves

AWS O PCS usa uma AWS chave KMS própria para criptografar dados. Para obter mais informações, consulte Chaves do cliente e AWS chaves no Guia do AWS KMS desenvolvedor. Você também pode usar uma chave gerenciada pelo cliente. Para obter mais informações, consulte Política de chave KMS necessária para uso com volumes criptografados do EBS no PCS AWS.

O segredo do cluster é armazenado AWS Secrets Manager e criptografado com a chave KMS gerenciada pelo Secrets Manager. Para obter mais informações, consulte Trabalhando com segredos de cluster no AWS PCS.

Privacidade do tráfego entre redes

AWS Os recursos de computação do PCS para um cluster residem em 1 VPC na conta do cliente. Portanto, todo o tráfego interno do serviço AWS PCS em um cluster permanece na AWS rede e não viaja pela Internet. A comunicação entre o usuário e os nós AWS PCS pode viajar pela Internet e recomendamos o uso de SSH ou Systems Manager para conectar-se aos nós. Para obter mais informações, consulte O que é AWS Systems Manager? no Guia do AWS Systems Manager usuário.

Você também pode usar as seguintes ofertas para conectar sua rede local a: AWS

- AWS Site-to-Site VPN. Para obter mais informações, consulte O que é AWS Site-to-Site VPN? no Guia do AWS Site-to-Site VPN usuário.
- Um AWS Direct Connect. Para obter mais informações, consulte O que é AWS Direct Connect? no Guia do AWS Direct Connect usuário.

Você acessa a API AWS PCS para realizar tarefas administrativas para o serviço. Você e seus usuários acessam as portas do endpoint do Slurm para interagir diretamente com o agendador.

Criptografia do tráfego da API

Para acessar a API AWS PCS, os clientes devem oferecer suporte ao Transport Layer Security (TLS) 1.2 ou posterior. Exigimos TLS 1.2 e recomendamos TLS 1.3. Os clientes também devem ter suporte a pacotes de criptografia com sigilo de encaminhamento perfeito (PFS) como Ephemeral Diffie-Hellman (DHE) ou Ephemeral Elliptic Curve Diffie-Hellman (ECDHE). A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos. Além disso, as solicitações devem ser assinadas usando um ID da chave de acesso e uma chave de acesso secreta associada a uma entidade principal do IAM. Você também pode usar AWS Security Token Service (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

Criptografia do tráfego de dados

A criptografia de dados em trânsito é habilitada a partir de EC2 instâncias compatíveis que acessam o endpoint do agendador e entre ComputeNodeGroup instâncias de dentro do. Nuvem AWS Para obter mais informações, consulte Criptografia em trânsito.

Política de chave KMS necessária para uso com volumes criptografados do EBS no PCS AWS

AWS O PCS usa funções vinculadas ao serviço para delegar permissões a outras pessoas. Serviços da AWS A função vinculada ao serviço AWS PCS é predefinida e inclui as permissões que o AWS PCS exige para ligar para outras pessoas Serviços da AWS em seu nome. As permissões predefinidas também incluem acesso às suas chaves gerenciadas pelo cliente Chaves gerenciadas pela AWS , mas não às suas.

Este tópico descreve como configurar a política de chaves necessária para iniciar instâncias quando você especifica uma chave gerenciada pelo cliente para a criptografia do Amazon EBS.

> **Note**
>
> AWS O PCS não exige autorização adicional para usar o padrão Chave gerenciada pela AWS para proteger os volumes criptografados em sua conta.

Conteúdo

Visão geral

Você pode usar o seguinte AWS KMS keys para a criptografia do Amazon EBS quando o AWS PCS inicia instâncias:

- Chave gerenciada pela AWS: uma chave de criptografia em sua conta que é criada por, pertencente a e gerenciada pelo Amazon EBS. Essa é a chave de criptografia padrão para uma nova conta. O Amazon EBS usa o Chave gerenciada pela AWS para criptografia, a menos que você especifique uma chave gerenciada pelo cliente.
- Chave gerenciada pelo cliente: uma chave de criptografia personalizada que você cria, possui e gerencia. Para obter mais informações, consulte Criar uma chave KMS no Guia do AWS Key Management Service desenvolvedor.

> **Note**
>
> A chave deve ser simétrica. O Amazon EBS não oferece suporte a chaves assimétricas gerenciadas pelo cliente.

Você configura as chaves gerenciadas pelo cliente ao criar instantâneos criptografados ou um modelo de execução que especifica volumes criptografados, ou quando opta por habilitar a criptografia por padrão.

Configurar políticas de chave

Suas chaves KMS devem ter uma política de chaves que permita ao AWS PCS iniciar instâncias com volumes do Amazon EBS criptografados com uma chave gerenciada pelo cliente.

Use os exemplos desta página para configurar uma política de chaves para dar ao AWS PCS acesso à sua chave gerenciada pelo cliente. Você pode modificar a política de chaves da chave gerenciada pelo cliente ao criar a chave ou posteriormente.

A política principal deve ter as seguintes declarações:

- Uma declaração que permite que a identidade do IAM especificada no `Principal` elemento use diretamente a chave gerenciada pelo cliente. Inclui permissões para realizar as `DescribeKey` operações AWS KMS `Encrypt` `DecryptReEncrypt*`,`GenerateDataKey*`,, e na chave.
- Uma declaração que permite que a identidade do IAM especificada no `Principal` elemento use a `CreateGrant` operação para gerar concessões que delegam um subconjunto de suas próprias permissões para aqueles Serviços da AWS que estão integrados com AWS KMS ou outro principal. Isso permite que eles usem a chave para criar recursos criptografados em seu nome.

Não altere nenhuma declaração existente na política ao adicionar as novas declarações de política à sua política principal.

Para obter mais informações, consulte:

- create-key na Referência de Comandos AWS CLI
- put-key-policy na AWS CLI Command Reference
- Encontre o ID da chave e o ARN da chave no Guia do desenvolvedor AWS Key Management Service
- Funções vinculadas a serviços para PCS AWS
- Criptografia do Amazon EBS no Guia do usuário do Amazon EBS
- AWS Key Management Serviceno Guia do AWS Key Management Service desenvolvedor

Exemplo 1: seções da política de chaves que permitem acesso à chave gerenciada pelo cliente

Adicione as seguintes declarações de política à política principal da chave gerenciada pelo cliente. Substitua o ARN de exemplo pelo ARN da sua função vinculada ao serviço. `AWSServiceRoleForPCS` Este exemplo de política dá à função vinculada ao serviço AWS PCS (`AWSServiceRoleForPCS`) permissões para usar a chave gerenciada pelo cliente.

```
{
```

```
    "Sid": "Allow service-linked role use of the customer managed key",
    "Effect": "Allow",
    "Principal": {
        "AWS": [
            "arn:aws:iam::account-id:role/aws-service-role/pcs.amazonaws.com/
AWSServiceRoleForPCS"
        ]
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
}
```

```
{
    "Sid": "Allow attachment of persistent resources",
    "Effect": "Allow",
    "Principal": {
        "AWS": [
            "arn:aws:iam::account-id:role/aws-service-role/pcs.amazonaws.com/
AWSServiceRoleForPCS"
        ]
    },
    "Action": [
        "kms:CreateGrant"
    ],
    "Resource": "*",
    "Condition": {
        "Bool": {
            "kms:GrantIsForAWSResource": true
        }
     }
}
```

Exemplo 2: seções da política de chaves que permitem acesso entre contas à chave gerenciada pelo cliente

Se você criar uma chave gerenciada pelo cliente em uma conta diferente da do cluster AWS PCS, deverá usar uma concessão em combinação com a política de chaves para permitir o acesso entre contas à chave.

Para conceder acesso à chave

1. Adicione as seguintes declarações de política à política de chaves da chave gerenciada pelo cliente. Substitua o ARN de exemplo pelo ARN da outra conta. *111122223333*Substitua pela ID da conta real na Conta da AWS qual você deseja criar o cluster AWS PCS. Isso permite que você conceda permissão para que um usuário ou uma função do IAM na conta especificada crie uma concessão para a chave usando o seguinte comando da CLI. Por padrão, os usuários não têm acesso à chave.

```
{.
   "Sid": "Allow external account 111122223333 use of the customer managed key",
   "Effect": "Allow",
   "Principal": {
       "AWS": [
           "arn:aws:iam::111122223333:root"
       ]
   },
   "Action": [
       "kms:Encrypt",
       "kms:Decrypt",
       "kms:ReEncrypt*",
       "kms:GenerateDataKey*",
       "kms:DescribeKey"
   ],
   "Resource": "*"
}
```

```
{
   "Sid": "Allow attachment of persistent resources in external
 account 111122223333",
   "Effect": "Allow",
   "Principal": {
       "AWS": [
           "arn:aws:iam::111122223333:root"
```

```
        ]
    },
    "Action": [
        "kms:CreateGrant"
    ],
    "Resource": "*"
}
```

2. Na conta na qual você deseja criar o cluster AWS PCS, crie uma concessão que delegue as permissões relevantes à função vinculada ao serviço AWS PCS. O valor de `grantee-principal` é o ARN da função vinculada ao serviço. O valor de `key-id` é o ARN da chave.

   O exemplo a seguir do comando da CLI create-grant fornece à função vinculada ao serviço `AWSServiceRoleForPCS` nomeada na *111122223333* conta permissões para usar a chave gerenciada pelo cliente na conta. *444455556666*

   ```
   aws kms create-grant \
     --region us-west-2 \
     --key-id arn:aws:kms:us-
   west-2:444455556666:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d \
     --grantee-principal arn:aws:iam::111122223333:role/aws-service-role/
   pcs.amazonaws.com/AWSServiceRoleForPCS \
     --operations "Encrypt" "Decrypt" "ReEncryptFrom" "ReEncryptTo" "GenerateDataKey"
    "GenerateDataKeyWithoutPlaintext" "DescribeKey" "CreateGrant"
   ```

   > **Note**
   >
   > O usuário que faz a solicitação deve ter permissões para usar a `kms:CreateGrant` ação.

   O exemplo de política do IAM a seguir permite que uma identidade do IAM (usuário ou função) na conta *111122223333* crie uma concessão para a chave gerenciada pelo cliente na conta*444455556666*.

   ```
   {
     "Version": "2012-10-17",
     "Statement": [
       {
         "Sid": "AllowCreationOfGrantForTheKMSKeyinExternalAccount444455556666",
   ```

```
      "Effect": "Allow",
      "Action": "kms:CreateGrant",
      "Resource": "arn:aws:kms:us-
west-2:444455556666:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    }
  ]
}
```

Para obter mais informações sobre como criar uma concessão para uma chave do KMS em uma Conta da AWS diferente, consulte Concessões no AWS KMS no Guia do desenvolvedor do AWS Key Management Service .

> **Important**
>
> O nome do perfil vinculado ao serviço especificado como a entidade principal do beneficiário deve ser o nome de um perfil existente. Depois de criar a concessão, para garantir que a concessão permita que o AWS PCS use a chave KMS especificada, não exclua e recrie a função vinculada ao serviço.

Editar políticas de chaves no console do AWS KMS

Os exemplos nas seções anteriores mostram apenas como adicionar declarações a uma política de chaves, que é apenas uma maneira de alterar uma política de chaves. A maneira mais fácil de alterar uma política de chaves é usar a visualização padrão do AWS KMS console para políticas de chaves e tornar uma identidade (usuário ou função) do IAM um dos principais usuários da política de chaves apropriada. Para obter mais informações, consulte Usando a visualização AWS Management Console padrão no Guia do AWS Key Management Service desenvolvedor.

> **Warning**
>
> As declarações de política de visualização padrão do console incluem permissões para realizar AWS KMS `Revoke` operações na chave gerenciada pelo cliente. Se você revogar uma concessão que deu Conta da AWS acesso a uma chave gerenciada pelo cliente em sua conta, os usuários dessa chave Conta da AWS perderão o acesso aos dados criptografados e à chave.

Acesso AWS Parallel Computing Service usando um endpoint de interface ()AWS PrivateLink

Você pode usar AWS PrivateLink para criar uma conexão privada entre sua VPC e AWS Parallel Computing Service ()AWS PCS. Você pode acessar AWS PCS como se estivesse em sua VPC, sem o uso de um gateway de internet, dispositivo NAT, conexão VPN ou conexão. AWS Direct Connect As instâncias na sua VPC não precisam de endereços IP públicos para acessar o AWS PCS.

Estabeleça essa conectividade privada criando um endpoint de interface, habilitado pelo AWS PrivateLink. Criaremos um endpoint de interface de rede em cada sub-rede que você habilitar para o endpoint de interface. Estas são interfaces de rede gerenciadas pelo solicitante que servem como ponto de entrada para o tráfego destinado ao AWS PCS.

Para obter mais informações, consulte Acesso Serviços da AWS por meio AWS PrivateLink do AWS PrivateLink Guia.

Considerações para AWS PCS

Antes de configurar um endpoint de interface para AWS PCS, consulte Acesse um serviço da AWS usando um endpoint VPC de interface no Guia.AWS PrivateLink

AWS PCS suporta fazer chamadas para todas as suas ações de API por meio do endpoint da interface.

Se sua VPC não tiver acesso direto à Internet, você deverá configurar um VPC endpoint para permitir que suas instâncias do grupo de nós de computação chamem a ação da API. AWS PCS `RegisterComputeNodeGroupInstance`

Crie um endpoint de interface para AWS PCS

Você pode criar um endpoint de interface para AWS PCS usar o console Amazon VPC ou AWS Command Line Interface o AWS CLI(). Para obter mais informações, consulte Criar um endpoint de interface no Guia do usuário do AWS PrivateLink .

Crie um endpoint de interface para AWS PCS usar o seguinte nome de serviço:

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

*region*Substitua pelo ID do Região da AWS para criar o endpoint, como`us-east-1`.

Se você habilitar o DNS privado para o endpoint da interface, poderá fazer solicitações de API a AWS PCS usando seu nome DNS regional padrão. Por exemplo, `.pcs.us-east-1.amazonaws.com`

Crie uma política de endpoint para seu endpoint de interface.

Uma política de endpoint é um recurso do IAM que você pode anexar ao endpoint de interface. A política de endpoint padrão permite acesso total AWS PCS por meio do endpoint da interface. Para controlar o acesso AWS PCS permitido pela sua VPC, anexe uma política de endpoint personalizada ao endpoint da interface.

Uma política de endpoint especifica as seguintes informações:

- As entidades principais que podem realizar ações (Contas da AWS, usuários do IAM e perfis do IAM).
- As ações que podem ser realizadas.
- Os recursos nos quais as ações podem ser executadas.

Para obter mais informações, consulte Controlar o acesso aos serviços usando políticas de endpoint no Guia do AWS PrivateLink .

Exemplo: política de VPC endpoint para ações AWS PCS

Veja a seguir um exemplo de uma política de endpoint personalizado. Quando você anexa essa política ao seu endpoint de interface, ela concede acesso às AWS PCS ações listadas para todos os principais do cluster com o especificado. *`cluster-id region`*Substitua pelo ID Região da AWS do cluster, como`us-east-1`. *`account-id`*Substitua pelo Conta da AWS número do cluster.

```
{
    "Statement": [
            {
                "Action": [
                "pcs:CreateCluster",
                "pcs:ListClusters",
                "pcs:DeleteCluster",
                "pcs:GetCluster",
                ],
                "Effect": "Allow",
                "Principal": "*",
                "Resource": [
```

```
                "arn:aws:pcs:region:account-id:cluster/cluster-id*"
            ]
        }
    ]
}
```

Identity and Access Management for AWS Parallel Computing Service

AWS Identity and Access Management (IAM) é uma ferramenta AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) para usar os recursos do AWS PCS. O IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Tópicos

Público

A forma como você usa AWS Identity and Access Management (IAM) difere, dependendo do trabalho que você faz no AWS PCS.

Usuário do serviço — Se você usar o serviço AWS PCS para fazer seu trabalho, seu administrador fornecerá as credenciais e as permissões necessárias. À medida que você usa mais recursos do

AWS PCS para fazer seu trabalho, talvez precise de permissões adicionais. Entender como o acesso é gerenciado pode ajudar a solicitar as permissões corretas ao administrador. Se você não conseguir acessar um recurso no AWS PCS, consulteSolução de problemas de identidade e acesso ao serviço de computação AWS paralela.

Administrador de serviços — Se você é responsável pelos recursos do AWS PCS em sua empresa, provavelmente tem acesso total ao AWS PCS. É seu trabalho determinar quais recursos e recursos do AWS PCS seus usuários do serviço devem acessar. Envie as solicitações ao administrador do IAM para alterar as permissões dos usuários de serviço. Revise as informações nesta página para compreender os conceitos básicos do IAM. Para saber mais sobre como sua empresa pode usar o IAM com o AWS PCS, consulteComo o serviço de computação AWS paralela funciona com o IAM.

Administrador do IAM — Se você for administrador do IAM, talvez queira saber detalhes sobre como criar políticas para gerenciar o acesso ao AWS PCS. Para ver exemplos de políticas baseadas em identidade do AWS PCS que você pode usar no IAM, consulte. Exemplos de políticas baseadas em identidade para o serviço de computação AWS paralela

Autenticação com identidades

A autenticação é a forma como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado (conectado AWS) como usuário do Usuário raiz da conta da AWS IAM ou assumindo uma função do IAM.

Você pode entrar AWS como uma identidade federada usando credenciais fornecidas por meio de uma fonte de identidade. AWS IAM Identity Center Usuários (IAM Identity Center), a autenticação de login único da sua empresa e suas credenciais do Google ou do Facebook são exemplos de identidades federadas. Quando você faz login como identidade federada, o administrador já configurou anteriormente a federação de identidades usando perfis do IAM. Ao acessar AWS usando a federação, você está assumindo indiretamente uma função.

Dependendo do tipo de usuário que você é, você pode entrar no AWS Management Console ou no portal de AWS acesso. Para obter mais informações sobre como fazer login AWS, consulte Como fazer login Conta da AWS no Guia do Início de Sessão da AWS usuário.

Se você acessar AWS programaticamente, AWS fornece um kit de desenvolvimento de software (SDK) e uma interface de linha de comando (CLI) para assinar criptograficamente suas solicitações usando suas credenciais. Se você não usa AWS ferramentas, você mesmo deve assinar as solicitações. Para obter mais informações sobre como usar o método recomendado para designar

solicitações por conta própria, consulte Versão 4 do AWS Signature para solicitações de API no Guia do usuário do IAM.

Independente do método de autenticação usado, também pode ser necessário fornecer informações adicionais de segurança. Por exemplo, AWS recomenda que você use a autenticação multifator (MFA) para aumentar a segurança da sua conta. Para saber mais, consulte Autenticação multifator no Guia do usuário do AWS IAM Identity Center e Usar a autenticação multifator da AWS no IAM no Guia do usuário do IAM.

Conta da AWS usuário root

Ao criar uma Conta da AWS, você começa com uma identidade de login que tem acesso completo a todos Serviços da AWS os recursos da conta. Essa identidade é chamada de usuário Conta da AWS raiz e é acessada fazendo login com o endereço de e-mail e a senha que você usou para criar a conta. É altamente recomendável não usar o usuário-raiz para tarefas diárias. Proteja as credenciais do usuário-raiz e use-as para executar as tarefas que somente ele puder executar. Para obter a lista completa das tarefas que exigem login como usuário-raiz, consulte Tarefas que exigem credenciais de usuário-raiz no Guia do Usuário do IAM.

Identidade federada

Como prática recomendada, exija que usuários humanos, incluindo usuários que precisam de acesso de administrador, usem a federação com um provedor de identidade para acessar Serviços da AWS usando credenciais temporárias.

Uma identidade federada é um usuário do seu diretório de usuários corporativo, de um provedor de identidade da web AWS Directory Service, do diretório do Identity Center ou de qualquer usuário que acesse usando credenciais fornecidas Serviços da AWS por meio de uma fonte de identidade. Quando as identidades federadas são acessadas Contas da AWS, elas assumem funções, e as funções fornecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, é recomendável usar o AWS IAM Identity Center. Você pode criar usuários e grupos no IAM Identity Center ou pode se conectar e sincronizar com um conjunto de usuários e grupos em sua própria fonte de identidade para uso em todos os seus Contas da AWS aplicativos. Para obter mais informações sobre o Centro de Identidade do IAM, consulte O que é o Centro de Identidade do IAM? no Guia do Usuário do AWS IAM Identity Center .

Usuários e grupos do IAM

Um usuário do IAM é uma identidade dentro da sua Conta da AWS que tem permissões específicas para uma única pessoa ou aplicativo. Sempre que possível, é recomendável contar com credenciais temporárias em vez de criar usuários do IAM com credenciais de longo prazo, como senhas e chaves de acesso. No entanto, se você tiver casos de uso específicos que exijam credenciais de longo prazo com usuários do IAM, é recomendável alternar as chaves de acesso. Para obter mais informações, consulte Alternar as chaves de acesso regularmente para casos de uso que exijam credenciais de longo prazo no Guia do Usuário do IAM.

Um grupo do IAM é uma identidade que especifica uma coleção de usuários do IAM. Não é possível fazer login como um grupo. É possível usar grupos para especificar permissões para vários usuários de uma vez. Os grupos facilitam o gerenciamento de permissões para grandes conjuntos de usuários. Por exemplo, você pode ter um grupo chamado IAMAdminse conceder a esse grupo permissões para administrar recursos do IAM.

Usuários são diferentes de perfis. Um usuário é exclusivamente associado a uma pessoa ou a uma aplicação, mas um perfil pode ser assumido por qualquer pessoa que precisar dele. Os usuários têm credenciais permanentes de longo prazo, mas os perfis fornecem credenciais temporárias. Para saber mais, consulte Casos de uso para usuários do IAM no Guia do usuário do IAM.

Perfis do IAM

Uma função do IAM é uma identidade dentro da sua Conta da AWS que tem permissões específicas. Ele é semelhante a um usuário do IAM, mas não está associado a uma pessoa específica. Para assumir temporariamente uma função do IAM no AWS Management Console, você pode alternar de um usuário para uma função do IAM (console). Você pode assumir uma função chamando uma operação de AWS API AWS CLI ou usando uma URL personalizada. Para obter mais informações sobre métodos para usar perfis, consulte Métodos para assumir um perfil no Guia do usuário do IAM.

Perfis do IAM com credenciais temporárias são úteis nas seguintes situações:

- Acesso de usuário federado: para atribuir permissões a identidades federadas, é possível criar um perfil e definir permissões para ele. Quando uma identidade federada é autenticada, essa identidade é associada ao perfil e recebe as permissões definidas por ele. Para ter mais informações sobre perfis para federação, consulte Criar um perfil para um provedor de identidade de terceiros (federação) no Guia do usuário do IAM. Se usar o Centro de Identidade do IAM, configure um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o Centro de Identidade do IAM correlaciona o conjunto de permissões a

um perfil no IAM. Para obter informações sobre conjuntos de permissões, consulte Conjuntos de Permissões no Guia do Usuário do AWS IAM Identity Center .

- Permissões temporárias para usuários do IAM: um usuário ou um perfil do IAM pode presumir um perfil do IAM para obter temporariamente permissões diferentes para uma tarefa específica.
- Acesso entre contas: é possível usar um perfil do IAM para permitir que alguém (uma entidade principal confiável) em outra conta acesse recursos em sua conta. Os perfis são a principal forma de conceder acesso entre contas. No entanto, com alguns Serviços da AWS, você pode anexar uma política diretamente a um recurso (em vez de usar uma função como proxy). Para conhecer a diferença entre perfis e políticas baseadas em recurso para acesso entre contas, consulte Acesso a recursos entre contas no IAM no Guia do usuário do IAM.
- Acesso entre serviços — Alguns Serviços da AWS usam recursos em outros Serviços da AWS. Por exemplo, quando você faz uma chamada em um serviço, é comum que esse serviço execute aplicativos na Amazon EC2 ou armazene objetos no Amazon S3. Um serviço pode fazer isso usando as permissões da entidade principal da chamada, usando um perfil de serviço ou um perfil vinculado ao serviço.
  - Sessões de acesso direto (FAS) — Quando você usa um usuário ou uma função do IAM para realizar ações AWS, você é considerado principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O FAS usa as permissões do diretor chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) para fazer solicitações aos serviços posteriores. As solicitações do FAS são feitas somente quando um serviço recebe uma solicitação que requer interações com outros Serviços da AWS ou com recursos para ser concluída. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte Sessões de acesso direto.
  - Perfil de serviço: um perfil de serviço é um perfil do IAM que um serviço assume para executar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte Criar um perfil para delegar permissões a um AWS service (Serviço da AWS) no Guia do Usuário do IAM.
  - Função vinculada ao serviço — Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um. AWS service (Serviço da AWS) O serviço pode presumir o perfil para executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para perfis vinculados a serviço.
- Aplicativos em execução na Amazon EC2 — Você pode usar uma função do IAM para gerenciar credenciais temporárias para aplicativos que estão sendo executados em uma EC2 instância e

fazendo solicitações AWS CLI de AWS API. Isso é preferível ao armazenamento de chaves de acesso na EC2 instância. Para atribuir uma AWS função a uma EC2 instância e disponibilizá-la para todos os aplicativos, você cria um perfil de instância anexado à instância. Um perfil de instância contém a função e permite que programas em execução na EC2 instância recebam credenciais temporárias. Para obter mais informações, consulte Usar uma função do IAM para conceder permissões a aplicativos executados em EC2 instâncias da Amazon no Guia do usuário do IAM.

Gerenciar o acesso usando políticas

Você controla o acesso AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política é um objeto AWS que, quando associada a uma identidade ou recurso, define suas permissões. AWS avalia essas políticas quando um principal (usuário, usuário raiz ou sessão de função) faz uma solicitação. As permissões nas políticas determinam se a solicitação será permitida ou negada. A maioria das políticas é armazenada AWS como documentos JSON. Para obter mais informações sobre a estrutura e o conteúdo de documentos de políticas JSON, consulte Visão geral das políticas JSON no Guia do usuário do IAM.

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

Por padrão, usuários e perfis não têm permissões. Para conceder permissão aos usuários para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM aos perfis e os usuários podem assumir os perfis.

As políticas do IAM definem permissões para uma ação independentemente do método usado para executar a operação. Por exemplo, suponha que você tenha uma política que permite a ação `iam:GetRole`. Um usuário com essa política pode obter informações de função da AWS Management Console AWS CLI, da ou da AWS API.

Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário, grupo de usuários ou perfil do IAM. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte Definir permissões personalizadas do IAM com as políticas gerenciadas pelo cliente no Guia do Usuário do IAM.

As políticas baseadas em identidade podem ser categorizadas como políticas em linha ou políticas gerenciadas. As políticas em linha são anexadas diretamente a um único usuário, grupo ou perfil. As políticas gerenciadas são políticas autônomas que você pode associar a vários usuários, grupos e funções em seu Conta da AWS. As políticas AWS gerenciadas incluem políticas gerenciadas e políticas gerenciadas pelo cliente. Para saber como escolher entre uma política gerenciada ou uma política em linha, consulte Escolher entre políticas gerenciadas e políticas em linha no Guia do usuário do IAM.

Políticas baseadas em recursos

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o atributo ao qual a política está anexada, a política define quais ações uma entidade principal especificado pode executar nesse atributo e em que condições. Você deve especificar uma entidade principal em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Políticas baseadas em recursos são políticas em linha localizadas nesse serviço. Você não pode usar políticas AWS gerenciadas do IAM em uma política baseada em recursos.

Listas de controle de acesso (ACLs)

As listas de controle de acesso (ACLs) controlam quais diretores (membros da conta, usuários ou funções) têm permissões para acessar um recurso. ACLs são semelhantes às políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

O Amazon S3 e o AWS WAF Amazon VPC são exemplos de serviços que oferecem suporte. ACLs Para saber mais ACLs, consulte a visão geral da lista de controle de acesso (ACL) no Guia do desenvolvedor do Amazon Simple Storage Service.

Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais menos comuns. Esses tipos de política podem definir o máximo de permissões concedidas a você pelos tipos de política mais comuns.

- Limites de permissões: um limite de permissões é um recurso avançado no qual você define o máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM (usuário ou perfil do IAM). É possível definir um limite de permissões para uma entidade.

As permissões resultantes são a interseção das políticas baseadas em identidade de uma entidade com seus limites de permissões. As políticas baseadas em recurso que especificam o usuário ou o perfil no campo `Principal` não são limitadas pelo limite de permissões. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações sobre limites de permissões, consulte Limites de permissões para identidades do IAM no Guia do usuário do IAM.

- Políticas de controle de serviço (SCPs) — SCPs são políticas JSON que especificam as permissões máximas para uma organização ou unidade organizacional (OU) em AWS Organizations. AWS Organizations é um serviço para agrupar e gerenciar centralmente vários Contas da AWS que sua empresa possui. Se você habilitar todos os recursos em uma organização, poderá aplicar políticas de controle de serviço (SCPs) a qualquer uma ou a todas as suas contas. O SCP limita as permissões para entidades nas contas dos membros, incluindo cada uma Usuário raiz da conta da AWS. Para obter mais informações sobre Organizations e SCPs, consulte Políticas de controle de serviços no Guia AWS Organizations do Usuário.
- Políticas de controle de recursos (RCPs) — RCPs são políticas JSON que você pode usar para definir o máximo de permissões disponíveis para recursos em suas contas sem atualizar as políticas do IAM anexadas a cada recurso que você possui. O RCP limita as permissões para recursos nas contas dos membros e pode afetar as permissões efetivas para identidades, incluindo a Usuário raiz da conta da AWS, independentemente de pertencerem à sua organização. Para obter mais informações sobre Organizations e RCPs, incluindo uma lista Serviços da AWS desse suporte RCPs, consulte Políticas de controle de recursos (RCPs) no Guia AWS Organizations do usuário.
- Políticas de sessão: são políticas avançadas que você transmite como um parâmetro quando cria de forma programática uma sessão temporária para um perfil ou um usuário federado. As permissões da sessão resultante são a interseção das políticas baseadas em identidade do usuário ou do perfil e das políticas de sessão. As permissões também podem ser provenientes de uma política baseada em recursos. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações, consulte Políticas de sessão no Guia do usuário do IAM.

Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como AWS determinar se uma solicitação deve ser permitida quando vários tipos de políticas estão envolvidos, consulte Lógica de avaliação de políticas no Guia do usuário do IAM.

Como o serviço de computação AWS paralela funciona com o IAM

Antes de usar o IAM para gerenciar o acesso ao AWS PCS, saiba quais recursos do IAM estão disponíveis para uso com o AWS PCS.

Recursos do IAM que você pode usar com o AWS Parallel Computing Service

Recurso do IAM	AWS Suporte para PCS
Políticas baseadas em identidade	Sim
Políticas baseadas em recurso	Não
Ações de políticas	Sim
Recursos de políticas	Sim
Chaves de condição de política (específicas do serviço)	Sim
ACLs	Não
ABAC (tags em políticas)	Sim
Credenciais temporárias	Sim
Permissões de entidade principal	Sim
Perfis de serviço	Não
Funções vinculadas ao serviço	Sim

Para ter uma visão de alto nível de como o AWS PCS e outros AWS serviços funcionam com a maioria dos recursos do IAM, consulte AWS os serviços que funcionam com o IAM no Guia do usuário do IAM.

Políticas baseadas em identidade para PCS AWS

Compatível com políticas baseadas em identidade: sim

As políticas baseadas em identidade são documentos de políticas de permissões JSON que você pode anexar a uma identidade, como usuário do IAM, grupo de usuários ou perfil. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte Definir permissões personalizadas do IAM com as políticas gerenciadas pelo cliente no Guia do Usuário do IAM.

Com as políticas baseadas em identidade do IAM, é possível especificar ações e recursos permitidos ou negados, assim como as condições sob as quais as ações são permitidas ou negadas. Você não pode especificar a entidade principal em uma política baseada em identidade porque ela se aplica ao usuário ou perfil ao qual ela está anexada. Para saber mais sobre todos os elementos que podem ser usados em uma política JSON, consulte Referência de elemento de política JSON do IAM no Guia do usuário do IAM.

Exemplos de políticas baseadas em identidade para PCS AWS

Para ver exemplos de políticas baseadas em identidade do AWS PCS, consulte. Exemplos de políticas baseadas em identidade para o serviço de computação AWS paralela

Políticas baseadas em recursos no PCS AWS

Compatibilidade com políticas baseadas em recursos: não

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o atributo ao qual a política está anexada, a política define quais ações uma entidade principal especificado pode executar nesse atributo e em que condições. Você deve especificar uma entidade principal em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Para permitir o acesso entre contas, você pode especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em recursos. Adicionar uma entidade principal entre contas à política baseada em recurso é apenas metade da tarefa de estabelecimento da relação de confiança. Quando o principal e o recurso são diferentes Contas da AWS, um administrador do IAM na conta confiável também deve conceder permissão à entidade principal (usuário ou função) para acessar o recurso. Eles concedem permissão ao anexar uma política baseada em identidade para a entidade. No entanto, se uma política baseada em recurso conceder acesso a uma entidade principal na mesma conta, nenhuma política baseada em

identidade adicional será necessária. Consulte mais informações em Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Ações políticas para AWS PCS

Compatível com ações de políticas: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento `Action` de uma política JSON descreve as ações que podem ser usadas para permitir ou negar acesso em uma política. As ações de política geralmente têm o mesmo nome da operação de AWS API associada. Existem algumas exceções, como ações somente de permissão, que não têm uma operação de API correspondente. Algumas operações também exigem várias ações em uma política. Essas ações adicionais são chamadas de ações dependentes.

Incluem ações em uma política para conceder permissões para executar a operação associada.

Para ver uma lista de ações do AWS PCS, consulte Ações definidas pelo serviço de computação AWS paralela na Referência de autorização de serviço.

As ações de política no AWS PCS usam o seguinte prefixo antes da ação:

```
pcs
```

Para especificar várias ações em uma única declaração, separe-as com vírgulas.

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

Recursos de políticas para AWS PCS

Compatível com recursos de políticas: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento de política JSON `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou `NotResource`. Como prática

recomendada, especifique um recurso usando seu nome do recurso da Amazon (ARN). Isso pode ser feito para ações que oferecem compatibilidade com um tipo de recurso específico, conhecido como permissões em nível de recurso.

Para ações que não oferecem compatibilidade com permissões em nível de recurso, como operações de listagem, use um curinga (*) para indicar que a instrução se aplica a todos os recursos.

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

Para ver uma lista dos tipos de recursos do AWS PCS e seus ARNs, consulte Recursos definidos pelo serviço de computação AWS paralela na Referência de autorização de serviço. Para saber com quais ações você pode especificar o ARN de cada recurso, consulte Ações definidas pelo serviço de computação AWS paralela.

Para ver exemplos de políticas baseadas em identidade do AWS PCS, consulte. Exemplos de políticas baseadas em identidade para o serviço de computação AWS paralela

Chaves de condição de política para AWS PCS

Compatível com chaves de condição de política específicas de serviço: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento `Condition` (ou bloco `Condition`) permite que você especifique condições nas quais uma instrução estiver em vigor. O elemento `Condition` é opcional. É possível criar expressões condicionais que usem agentes de condição, como “igual a” ou “menor que”, para fazer a condição da política corresponder aos valores na solicitação.

Se você especificar vários elementos de `Condition` em uma declaração ou várias chaves em um único elemento de `Condition`, a AWS os avaliará usando uma operação lógica `AND`. Se você especificar vários valores para uma única chave de condição, AWS avalia a condição usando uma `OR` operação lógica. Todas as condições devem ser atendidas antes que as permissões da instrução sejam concedidas.

Você também pode usar variáveis de espaço reservado ao especificar condições. Por exemplo, é possível conceder a um usuário do IAM permissão para acessar um recurso somente se ele estiver

marcado com seu nome de usuário do IAM. Para obter mais informações, consulte Elementos da política do IAM: variáveis e tags no Guia do usuário do IAM.

AWS suporta chaves de condição globais e chaves de condição específicas do serviço. Para ver todas as chaves de condição AWS globais, consulte as chaves de contexto de condição AWS global no Guia do usuário do IAM.

Para ver uma lista das chaves de condição do AWS PCS, consulte Chaves de condição para o serviço de computação AWS paralela na Referência de autorização de serviço. Para saber com quais ações e recursos você pode usar uma chave de condição, consulte Ações definidas pelo serviço de computação AWS paralela.

Para ver exemplos de políticas baseadas em identidade do AWS PCS, consulte. Exemplos de políticas baseadas em identidade para o serviço de computação AWS paralela

ACLs em AWS PCS

Suportes ACLs: Não

As listas de controle de acesso (ACLs) controlam quais diretores (membros da conta, usuários ou funções) têm permissões para acessar um recurso. ACLs são semelhantes às políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

ABAC com AWS PCS

Compatível com ABAC (tags em políticas): sim

O controle de acesso por atributo (ABAC) é uma estratégia de autorização que define as permissões com base em atributos. Em AWS, esses atributos são chamados de tags. Você pode anexar tags a entidades do IAM (usuários ou funções) e a vários AWS recursos. Marcar de entidades e atributos é a primeira etapa do ABAC. Em seguida, você cria políticas de ABAC para permitir operações quando a tag da entidade principal corresponder à tag do recurso que ela estiver tentando acessar.

O ABAC é útil em ambientes que estão crescendo rapidamente e ajuda em situações em que o gerenciamento de políticas se torna um problema.

Para controlar o acesso baseado em tags, forneça informações sobre as tags no elemento de condição de uma política usando as `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* ou chaves de condição `aws:TagKeys`.

Se um serviço for compatível com as três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço for compatível com as três chaves de condição somente para alguns tipos de recursos, o valor será Parcial

Para obter mais informações sobre o ABAC, consulte Definir permissões com autorização do ABAC no Guia do usuário do IAM. Para visualizar um tutorial com etapas para configurar o ABAC, consulte Usar controle de acesso baseado em atributos (ABAC) no Guia do usuário do IAM.

Usando credenciais temporárias com AWS o PCS

Compatível com credenciais temporárias: sim

Alguns Serviços da AWS não funcionam quando você faz login usando credenciais temporárias. Para obter informações adicionais, incluindo quais Serviços da AWS funcionam com credenciais temporárias, consulte Serviços da AWS "Trabalhe com o IAM" no Guia do usuário do IAM.

Você está usando credenciais temporárias se fizer login AWS Management Console usando qualquer método, exceto um nome de usuário e senha. Por exemplo, quando você acessa AWS usando o link de login único (SSO) da sua empresa, esse processo cria automaticamente credenciais temporárias. Você também cria automaticamente credenciais temporárias quando faz login no console como usuário e, em seguida, alterna perfis. Para obter mais informações sobre como alternar funções, consulte Alternar para um perfil do IAM (console) no Guia do usuário do IAM.

Você pode criar manualmente credenciais temporárias usando a AWS API AWS CLI ou. Em seguida, você pode usar essas credenciais temporárias para acessar AWS. AWS recomenda que você gere credenciais temporárias dinamicamente em vez de usar chaves de acesso de longo prazo. Para obter mais informações, consulte Credenciais de segurança temporárias no IAM.

Permissões principais entre serviços para AWS PCS

Compatibilidade com o recurso de encaminhamento de sessões de acesso (FAS): sim

Quando você usa um usuário ou uma função do IAM para realizar ações AWS, você é considerado principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. O FAS usa as permissões do diretor chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) para fazer solicitações aos serviços posteriores. As solicitações do FAS são feitas somente quando um serviço recebe uma solicitação que requer interações com outros Serviços da AWS ou com recursos para ser concluída. Nesse caso, você precisa ter permissões para executar ambas as ações. Para obter detalhes da política ao fazer solicitações de FAS, consulte Sessões de acesso direto.

Funções de serviço para AWS PCS

Compatível com perfis de serviço: não

O perfil de serviço é um perfil do IAM que um serviço assume para executar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para obter mais informações, consulte Criar um perfil para delegar permissões a um AWS service (Serviço da AWS) no Guia do Usuário do IAM.

> **Warning**
>
> Alterar as permissões de uma função de serviço pode interromper a funcionalidade do AWS PCS. Edite as funções de serviço somente quando o AWS PCS fornecer orientação para fazer isso.

Funções vinculadas a serviços para PCS AWS

Compatibilidade com perfis vinculados a serviços: sim

Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um. AWS service (Serviço da AWS) O serviço pode presumir o perfil para executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para perfis vinculados a serviço.

Para obter detalhes sobre como criar ou gerenciar funções vinculadas ao serviço AWS PCS, consulte. Funções vinculadas a serviços para PCS AWS

Exemplos de políticas baseadas em identidade para o serviço de computação AWS paralela

Por padrão, usuários e funções não têm permissão para criar ou modificar recursos do AWS PCS. Eles também não podem realizar tarefas usando a AWS API AWS Management Console, AWS Command Line Interface (AWS CLI) ou. Para conceder permissão aos usuários para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM. O administrador pode então adicionar as políticas do IAM aos perfis e os usuários podem assumir os perfis.

Para aprender a criar uma política baseada em identidade do IAM ao usar esses documentos de política em JSON de exemplo, consulte Criar políticas do IAM (console) no Guia do usuário do IAM.

Para obter detalhes sobre ações e tipos de recursos definidos pelo AWS PCS, incluindo o formato do ARNs para cada um dos tipos de recursos, consulte Ações, recursos e chaves de condição para o serviço de computação AWS paralela na Referência de autorização de serviço.

Tópicos

- Práticas recomendadas de política
- Usando o console AWS PCS
- Permitir que os usuários visualizem suas próprias permissões

Práticas recomendadas de política

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir recursos do AWS PCS em sua conta. Essas ações podem incorrer em custos para sua Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas AWS gerenciadas e avance para as permissões de privilégios mínimos — Para começar a conceder permissões aos seus usuários e cargas de trabalho, use as políticas AWS gerenciadas que concedem permissões para muitos casos de uso comuns. Eles estão disponíveis no seu Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo AWS cliente que sejam específicas para seus casos de uso. Para obter mais informações, consulte Políticas gerenciadas pela AWS ou Políticas gerenciadas pela AWS para funções de trabalho no Guia do usuário do IAM.
- Aplique permissões de privilégio mínimo: ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em recursos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para obter mais informações sobre como usar o IAM para aplicar permissões, consulte Políticas e permissões no IAM no Guia do usuário do IAM.
- Use condições nas políticas do IAM para restringir ainda mais o acesso: você pode adicionar uma condição às políticas para limitar o acesso a ações e recursos. Por exemplo, você pode escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso às ações de serviço se elas forem usadas por meio de uma ação específica AWS service (Serviço da AWS), como AWS CloudFormation. Para obter mais informações, consulte Elementos da política JSON do IAM: condição no Guia do usuário do IAM.

- Use o IAM Access Analyzer para validar suas políticas do IAM a fim de garantir permissões seguras e funcionais: o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam a linguagem de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de cem verificações de política e recomendações práticas para ajudar a criar políticas seguras e funcionais. Para obter mais informações, consulte Validação de políticas do IAM Access Analyzer no Guia do Usuário do IAM.
- Exigir autenticação multifator (MFA) — Se você tiver um cenário que exija usuários do IAM ou um usuário root, ative Conta da AWS a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para obter mais informações, consulte Configuração de acesso à API protegido por MFA no Guia do Usuário do IAM.

Para obter mais informações sobre as práticas recomendadas do IAM, consulte Práticas recomendadas de segurança no IAM no Guia do usuário do IAM.

Usando o console AWS PCS

Para acessar o console do AWS Parallel Computing Service, você deve ter um conjunto mínimo de permissões. Essas permissões devem permitir que você liste e visualize detalhes sobre os recursos do AWS PCS em seu Conta da AWS. Caso crie uma política baseada em identidade mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para entidades (usuários ou perfis) com essa política.

Você não precisa permitir permissões mínimas do console para usuários que estão fazendo chamadas somente para a API AWS CLI ou para a AWS API. Em vez disso, permita o acesso somente a ações que correspondam à operação de API que estiverem tentando executar.

Para obter mais informações sobre as permissões mínimas necessárias para usar o console AWS PCS, consultePermissões mínimas para AWS PCS.

Permitir que os usuários visualizem suas próprias permissões

Este exemplo mostra como criar uma política que permita que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou programaticamente usando a API AWS CLI ou AWS .

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS políticas gerenciadas para o Serviço de Computação AWS Paralela

Uma política AWS gerenciada é uma política autônoma criada e administrada por AWS. AWS as políticas gerenciadas são projetadas para fornecer permissões para muitos casos de uso comuns, para que você possa começar a atribuir permissões a usuários, grupos e funções.

Lembre-se de que as políticas AWS gerenciadas podem não conceder permissões de privilégio mínimo para seus casos de uso específicos porque estão disponíveis para uso de todos os AWS clientes. Recomendamos que você reduza ainda mais as permissões definindo as políticas gerenciadas pelo cliente que são específicas para seus casos de uso.

Você não pode alterar as permissões definidas nas políticas AWS gerenciadas. Se AWS atualizar as permissões definidas em uma política AWS gerenciada, a atualização afetará todas as identidades principais (usuários, grupos e funções) às quais a política está anexada. AWS é mais provável que atualize uma política AWS gerenciada quando uma nova AWS service (Serviço da AWS) é lançada ou novas operações de API são disponibilizadas para serviços existentes.

Para mais informações, consulte Políticas gerenciadas pela AWS no Manual do usuário do IAM.

AWS política gerenciada: AWSPCSCompute NodePolicy

Você pode anexar AWSPCSCompute NodePolicy às suas entidades do IAM. Você pode anexar essa política a uma função do IAM do nó de computação do AWS PCS que você especifica para permitir que os nós que usam essa função se conectem a um cluster do AWS PCS.

AWS O PCS associa essa política a uma função de grupo de nós de computação quando você usa o console para criar um grupo de nós de computação.

Detalhes das permissões

Esta política inclui as seguintes permissões.

- `pcs:RegisterComputeNodeGroupInstance`— Permitir que um nó de computação AWS PCS (EC2 instância) se registre em um cluster AWS PCS.

Para visualizar as permissões para esta política, consulte AWSPCSComputeNodePolicy na Referência de políticas gerenciadas pela AWS .

AWS política gerenciada: AWSPCSService RolePolicy

Você não pode se vincular AWSPCSService RolePolicy às suas entidades do IAM. Essa política está vinculada a uma função vinculada ao serviço que permite que o AWS PCS execute ações em seu nome. Para obter mais informações, consulte Funções vinculadas a serviços para PCS AWS.

Detalhes das permissões

Esta política inclui as seguintes permissões.

- `ec2`— Permite que o AWS PCS crie e gerencie EC2 recursos da Amazon.
- `iam`— Permite que a AWS PCS crie uma função vinculada a serviços para a EC2 frota da Amazon e passe a função para a Amazon. EC2
- `cloudwatch`— Permite que a AWS PCS publique métricas de serviço na Amazon CloudWatch.
- `secretsmanager`— Permite que o AWS PCS gerencie segredos dos recursos do cluster AWS PCS.

Para visualizar as permissões para esta política, consulte AWSPCSServiceRolePolicy na Referência de políticas gerenciadas pela AWS .

AWS Atualizações do PCS para políticas AWS gerenciadas

Veja detalhes sobre as atualizações das políticas AWS gerenciadas do AWS PCS desde que esse serviço começou a rastrear essas alterações. Para alertas automáticos sobre alterações nesta página, assine o feed RSS na página de histórico de documentos do AWS PCS.

Alteração	Descrição	Data
AWSPCSComputeNodePolicy – Nova política	AWS O PCS adicionou uma nova política para conceder permissão aos nós de computação do AWS PCS para se conectarem aos clusters do AWS PCS. AWS O PCS associa essa política a uma função do IAM	23 de junho de 2025

Alteração	Descrição	Data
	quando você cria um grupo de nós de computação no console do AWS PCS.	
Atualizou o JSON neste documento	Foi corrigido o JSON neste documento para incluir. `"arn:aws:ec2:*:*:spot-instances-request/*"`	5 de setembro de 2024
AWS O PCS começou a rastrear as mudanças	AWS A PCS começou a monitorar as mudanças em suas políticas AWS gerenciadas.	28 de agosto de 2024

Funções vinculadas a serviços para PCS AWS

AWS O Parallel Computing Service usa AWS Identity and Access Management funções vinculadas ao serviço (IAM). Uma função vinculada ao serviço é um tipo exclusivo de função do IAM vinculada diretamente ao AWS PCS. As funções vinculadas ao serviço são predefinidas pelo AWS PCS e incluem todas as permissões que o serviço exige para chamar outros AWS serviços em seu nome.

Uma função vinculada ao serviço facilita a configuração do AWS PCS porque você não precisa adicionar manualmente as permissões necessárias. AWS O PCS define as permissões de suas funções vinculadas ao serviço e, a menos que definido de outra forma, somente o AWS PCS pode assumir suas funções. As permissões definidas incluem as políticas de confiança e de permissões, e essa política de permissões não pode ser anexada a nenhuma outra entidade do IAM.

Você só pode excluir um perfil vinculado a serviço depois de excluir os recursos relacionados. Isso protege seus recursos do AWS PCS porque você não pode remover acidentalmente a permissão para acessar os recursos.

Para obter informações sobre outros serviços que oferecem suporte a funções vinculadas a serviços, consulte AWS Serviços que funcionam com IAM e procure os serviços que têm Sim na coluna Funções vinculadas ao serviço. Escolha um Sim com um link para visualizar a documentação do perfil vinculado a esse serviço.

Permissões de função vinculadas ao serviço para PCS AWS

AWS O PCS usa a função vinculada ao serviço chamada AWSServiceRoleForPCS — Concede permissão ao AWS PCS para gerenciar recursos da Amazon EC2 .

A função vinculada ao serviço AWSService RoleFor PCS confia nos seguintes serviços para assumir a função:

- `pcs.amazonaws.com`

A política de permissões de função nomeada AWSPCSServiceRolePolicypermite que o AWS PCS conclua ações em recursos específicos.

Você deve configurar permissões para permitir que seus usuários, grupos ou perfis criem, editem ou excluam um perfil vinculado ao serviço. Para obter mais informações, consulte Service-linked role permissions (Permissões de nível vinculado a serviços) no Guia do usuário do IAM.

Criação de uma função vinculada a serviços para PCS AWS

Você não precisa criar manualmente uma função vinculada ao serviço. AWS O PCS cria uma função vinculada ao serviço para você quando você cria um cluster.

Editando uma função vinculada ao serviço para PCS AWS

AWS O PCS não permite que você edite a função vinculada ao serviço AWSService RoleFor PCS. Depois que criar um perfil vinculado ao serviço, você não poderá alterar o nome do perfil, pois várias entidades podem fazer referência a ele. No entanto, será possível editar a descrição do perfil usando o IAM. Para obter mais informações, consulte Editar um perfil vinculado ao serviço no Guia do usuário do IAM.

Excluindo uma função vinculada ao serviço para PCS AWS

Se você não precisar mais usar um recurso ou serviço que requer um perfil vinculado ao serviço, é recomendável excluí-lo. Dessa forma, você não tem uma entidade não utilizada que não seja monitorada ativamente ou mantida. No entanto, você deve limpar os recursos de seu perfil vinculado ao serviço antes de excluí-lo manualmente.

> **Note**
>
> Se o serviço AWS PCS estiver usando a função quando você tentar excluir os recursos, a exclusão poderá falhar. Se isso acontecer, espere alguns minutos e tente a operação novamente.

Para remover recursos do AWS PCS usados pelo AWSService RoleFor PCS

Você deve excluir todos os seus clusters para excluir a função vinculada ao serviço AWSService RoleFor PCS. Para obter mais informações, consulte Excluir um cluster.

Como excluir manualmente o perfil vinculado ao serviço usando o IAM

Use o console do IAM AWS CLI, o ou a AWS API para excluir a função vinculada ao serviço AWSService RoleFor PCS. Para obter mais informações, consulte Excluir um perfil vinculado ao serviço no Guia do usuário do IAM.

Regiões suportadas para funções vinculadas ao serviço AWS PCS

AWS O PCS oferece suporte ao uso de funções vinculadas a serviços em todas as regiões em que o serviço está disponível. Para obter mais informações, consulte Regiões e endpoints da AWS.

Função do Amazon EC2 Spot para AWS PCS

Se você quiser criar um grupo de nós de computação AWS PCS que use o Spot como opção de compra, você também deve ter a função vinculada ao serviço AWSServiceRoleForEC2Spot em seu. Conta da AWS Você pode usar o AWS CLI comando a seguir para criar a função. Para obter mais informações, consulte Criar uma função vinculada ao serviço e Criar uma função para delegar permissões a um AWS serviço no Guia do AWS Identity and Access Management usuário.

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

> **Note**
>
> Você receberá o seguinte erro se Conta da AWS já tiver uma função `AWSServiceRoleForEC2Spot` do IAM.

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

Permissões mínimas para AWS PCS

Esta seção descreve as permissões mínimas do IAM necessárias para que uma identidade do IAM (usuário, grupo ou função) use o serviço.

Sumário

Permissões mínimas para usar ações de API

Ação da API	Permissões mínimas	Permissões adicionais para o console
CreateCluster	`ec2:CreateNetworkInterface, ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:GetSecurityGroupsForVpc, iam:CreateServiceLinkedRole, secretsmanager:CreateSecret, secretsmanager:TagResource, pcs:CreateCluster`	

Ação da API	Permissões mínimas	Permissões adicionais para o console
ListClusters	`pcs:ListClusters`	
GetCluster	`pcs:GetCluster`	`ec2:DescribeSubnets`
DeleteCluster	`pcs:DeleteCluster`	
CreateComputeNodeGroup	`ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:DescribeLaunchTemplates, ec2:DescribeLaunchTemplateVersions, ec2:DescribeInstanceTypes, ec2:DescribeInstanceTypeOfferings, 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`

Ação da API	Permissões mínimas	Permissões adicionais para o console
UpdateComputeNodeGroup	`ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:DescribeLaunchTemplates, ec2:DescribeLaunchTemplateVersions, ec2:DescribeInstanceTypes, ec2:DescribeInstanceTypeOfferings, ec2:RunInstances, ec2:CreateFleet, ec2:CreateTags, iam:PassRole, iam:GetInstanceProfile, pcs:UpdateComputeNodeGroup`	`pcs:GetComputeNodeGroup, iam:ListInstanceProfiles, ec2:DescribeImages, pcs:GetCluster`
DeleteComputeNodeGroup	`pcs:DeleteComputeNodeGroup`	
CreateQueue	`pcs:CreateQueue`	`pcs:ListComputeNodeGroups, pcs:GetCluster`
ListQueues	`pcs:ListQueues`	`pcs:GetCluster`
GetQueue	`pcs:GetQueue`	

Ação da API	Permissões mínimas	Permissões adicionais para o console
UpdateQueue	`pcs:UpdateQueue`	`pcs:ListComputeNodeGroups, pcs:GetQueue`
DeleteQueue	`pcs:DeleteQueue`	

Permissões mínimas para usar tags

As permissões a seguir são necessárias para usar tags com seus recursos no AWS PCS.

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

Permissões mínimas para suportar registros

AWS O PCS envia dados de log para o Amazon CloudWatch Logs (CloudWatch Logs). Você deve garantir que sua identidade tenha as permissões mínimas para usar o CloudWatch Logs. Para obter mais informações, consulte Visão geral do gerenciamento de permissões de acesso aos seus recursos de CloudWatch registros no Guia do usuário do Amazon CloudWatch Logs.

Para obter informações sobre as permissões necessárias para que um serviço envie CloudWatch registros para o Logs, consulte Habilitar o registro de AWS serviços no Guia do usuário do Amazon CloudWatch Logs.

Permissões mínimas para um administrador de serviços

A política do IAM a seguir especifica as permissões mínimas necessárias para que uma identidade do IAM (usuário, grupo ou função) configure e gerencie o serviço AWS PCS.

> **Note**
>
> Os usuários que não configuram e gerenciam o serviço não precisam dessas permissões. Os usuários que executam apenas trabalhos usam o secure shell (SSH) para se conectar

ao cluster. AWS Identity and Access Management (IAM) não lida com autenticação ou autorização para SSH.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PCSAccess",
      "Effect": "Allow",
      "Action": [
        "pcs:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "EC2Access",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface",
        "ec2:DescribeImages",
        "ec2:GetSecurityGroupsForVpc",
        "ec2:DescribeSubnets",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeVpcs",
        "ec2:DescribeLaunchTemplates",
        "ec2:DescribeLaunchTemplateVersions",
        "ec2:DescribeInstanceTypes",
        "ec2:DescribeInstanceTypeOfferings",
        "ec2:RunInstances",
        "ec2:CreateFleet",
        "ec2:CreateTags"
      ],
      "Resource": "*"
    },
    {
      "Sid": "IamInstanceProfile",
      "Effect": "Allow",
      "Action": [
        "iam:GetInstanceProfile"
      ],
      "Resource": "*"
    },
```

```
    {
      "Sid": "IamPassRole",
      "Effect": "Allow",
      "Action": [
        "iam:PassRole"
      ],
      "Resource": [
        "arn:aws:iam::*:role/*/AWSPCS*",
        "arn:aws:iam::*:role/AWSPCS*",
        "arn:aws:iam::*:role/aws-pcs/*",
        "arn:aws:iam::*:role/*/aws-pcs/*"
      ],
      "Condition": {
        "StringEquals": {
           "iam:PassedToService": [
             "ec2.amazonaws.com"
           ]
        }
      }
    },
    {
      "Sid": "SLRAccess",
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
      "Resource": [
        "arn:aws:iam::*:role/aws-service-role/pcs.amazonaws.com/AWSServiceRoleFor*",
        "arn:aws:iam::*:role/aws-service-role/spot.amazonaws.com/AWSServiceRoleFor*"
      ],
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": [
            "pcs.amazonaws.com",
            "spot.amazonaws.com"
          ]
        }
      }
    },
    {
      "Sid": "AccessKMSKey",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
```

```
        "kms:Encrypt",
        "kms:GenerateDataKey",
        "kms:CreateGrant",
        "kms:DescribeKey"
      ],
      "Resource": "*"
    },
    {
      "Sid": "SecretManagementAccess",
      "Effect": "Allow",
      "Action": [
        "secretsmanager:CreateSecret",
        "secretsmanager:TagResource",
        "secretsmanager:UpdateSecret"
      ],
      "Resource": "*"
    },
    {
       "Sid": "ServiceLogsDelivery",
       "Effect": "Allow",
       "Action": [
         "pcs:AllowVendedLogDeliveryForResource",
         "logs:PutDeliverySource",
         "logs:PutDeliveryDestination",
         "logs:CreateDelivery"
       ],
       "Resource": "*"
    }
  ]
}
```

Perfis de instância do IAM para o AWS Parallel Computing Service

Os aplicativos executados em uma EC2 instância devem incluir AWS credenciais em todas as solicitações de AWS API feitas. Recomendamos que você use uma função do IAM para gerenciar credenciais temporárias na EC2 instância. Você pode definir um perfil de instância para fazer isso e anexá-lo às suas instâncias. Para obter mais informações, consulte as funções do IAM para a Amazon EC2 no Guia do usuário do Amazon Elastic Compute Cloud.

> **Note**
>
> Quando você usa o AWS Management Console para criar uma função do IAM para a Amazon EC2, o console cria um perfil de instância automaticamente e dá a ele o mesmo nome da função do IAM. Se você usa as AWS CLI ações de AWS API ou um AWS SDK para criar a função do IAM, você cria o perfil da instância como uma ação separada. Para obter mais informações, consulte Perfis de instância no Guia do usuário do Amazon Elastic Compute Cloud.

Você deve especificar o Amazon Resource Name (ARN) de um perfil de instância ao criar grupos de nós de computação. Você pode escolher perfis de instância diferentes para alguns ou todos os grupos de nós de computação.

Requisitos de perfil de instância

ARN do perfil da instância

A parte do nome da função do IAM do ARN deve começar com `AWSPCS` ou conter `/aws-pcs/` em seu caminho:

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

> **Note**
>
> Se você usar o AWS CLI, forneça um `--path` valor `iam create-instance-profile` a ser incluído `/aws-pcs/` no caminho do ARN. Por exemplo:
>
> ```
> aws iam create-instance-profile --path /aws-pcs/ --instance-profile-name
>  example-role-2
> ```

Permissões

No mínimo, o perfil da instância do AWS PCS deve incluir a política a seguir. Ele permite que os nós de computação notifiquem o serviço AWS PCS quando estiverem operacionais.

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

Políticas adicionais

Você pode considerar adicionar políticas gerenciadas ao perfil da instância. Por exemplo:

- O AmazonS3 ReadOnlyAccess fornece acesso somente de leitura a todos os buckets do S3.
- A Amazon SSMManaged InstanceCore habilita a funcionalidade principal do serviço AWS Systems Manager, como acesso remoto diretamente do Amazon Management Console.
- CloudWatchAgentServerPolicycontém as permissões necessárias para uso AmazonCloudWatchAgent em servidores.

Você também pode incluir suas próprias políticas de IAM que ofereçam suporte ao seu caso de uso específico.

Criar um perfil da instância

Para criar um perfil de instância, você pode:

- Selecione Criar um perfil básico ao criar um grupo de nós de computação para que o AWS PCS crie um para você com a política mínima exigida.
- Crie um perfil de instância diretamente do EC2 console da Amazon. Para obter mais informações, consulte Como usar perfis de instância no Guia AWS Identity and Access Management do usuário.

Listar perfis de instância para AWS PCS

Você pode usar o AWS CLI comando a seguir para listar os perfis de instância em um Região da AWS que atenda aos requisitos de nome do AWS PCS. *us-east-1*Substitua pelo apropriado Região da AWS.

```
aws iam list-instance-profiles --region us-east-1 --query "InstanceProfiles[?
starts_with(InstanceProfileName, 'AWSPCS') || contains(Path, '/aws-pcs/')].
[InstanceProfileName]" --output text
```

Solução de problemas de identidade e acesso ao serviço de computação AWS paralela

Use as informações a seguir para ajudá-lo a diagnosticar e corrigir problemas comuns que você pode encontrar ao trabalhar com o AWS PCS e o IAM.

Tópicos

- Não estou autorizado a realizar uma ação no AWS PCS
- Não estou autorizado a realizar iam: PassRole
- Quero permitir que pessoas fora da minha acessem meus Conta da AWS recursos do AWS PCS

Não estou autorizado a realizar uma ação no AWS PCS

Se você receber uma mensagem de erro informando que não tem autorização para executar uma ação, suas políticas deverão ser atualizadas para permitir que você realize a ação.

O erro do exemplo a seguir ocorre quando o usuário do IAM `mateojackson` tenta usar o console para visualizar detalhes sobre um atributo *my-example-widget* fictício, mas não tem as permissões `pcs:`*GetWidget* fictícias.

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

Nesse caso, a política do usuário `mateojackson` deve ser atualizada para permitir o acesso ao recurso *my-example-widget* usando a ação `pcs:`*GetWidget*.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Não estou autorizado a realizar iam: PassRole

Se você receber um erro informando que não está autorizado a realizar a `iam:PassRole` ação, suas políticas devem ser atualizadas para permitir que você passe uma função para o AWS PCS.

Alguns Serviços da AWS permitem que você passe uma função existente para esse serviço em vez de criar uma nova função de serviço ou uma função vinculada ao serviço. Para fazer isso, é preciso ter permissões para passar o perfil para o serviço.

O exemplo de erro a seguir ocorre quando um usuário do IAM chamado `marymajor` tenta usar o console para realizar uma ação no AWS PCS. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

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

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Quero permitir que pessoas fora da minha acessem meus Conta da AWS recursos do AWS PCS

Você pode criar um perfil que os usuários de outras contas ou pessoas fora da organização podem usar para acessar seus recursos. É possível especificar quem é confiável para assumir o perfil. Para serviços que oferecem suporte a políticas baseadas em recursos ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se o AWS PCS oferece suporte a esses recursos, consulteComo o serviço de computação AWS paralela funciona com o IAM.
- Para saber como fornecer acesso aos seus recursos em todos os Contas da AWS que você possui, consulte Como fornecer acesso a um usuário do IAM em outro Conta da AWS que você possui no Guia do usuário do IAM.
- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte Como fornecer acesso Contas da AWS a terceiros no Guia do usuário do IAM.

- Para saber como conceder acesso por meio da federação de identidades, consulte Conceder acesso a usuários autenticados externamente (federação de identidades) no Guia do usuário do IAM.
- Para saber a diferença entre perfis e políticas baseadas em recurso para acesso entre contas, consulte Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Validação de conformidade para o serviço de computação AWS paralela

Para saber se um AWS service (Serviço da AWS) está dentro do escopo de programas de conformidade específicos, consulte Serviços da AWS Escopo por Programa de Conformidade Serviços da AWS e escolha o programa de conformidade em que você está interessado. Para obter informações gerais, consulte Programas de AWS conformidade Programas AWS de .

Você pode baixar relatórios de auditoria de terceiros usando AWS Artifact. Para obter mais informações, consulte Baixar relatórios em AWS Artifact .

Sua responsabilidade de conformidade ao usar Serviços da AWS é determinada pela confidencialidade de seus dados, pelos objetivos de conformidade de sua empresa e pelas leis e regulamentações aplicáveis. AWS fornece os seguintes recursos para ajudar na conformidade:

- Governança e conformidade de segurança: esses guias de implementação de solução abordam considerações sobre a arquitetura e fornecem etapas para implantar recursos de segurança e conformidade.
- Referência de serviços qualificados para HIPAA: lista os serviços qualificados para HIPAA. Nem todos Serviços da AWS são elegíveis para a HIPAA.
- AWS Recursos de https://aws.amazon.com/compliance/resources/ de conformidade — Essa coleção de pastas de trabalho e guias pode ser aplicada ao seu setor e local.
- AWS Guias de conformidade do cliente — Entenda o modelo de responsabilidade compartilhada sob a ótica da conformidade. Os guias resumem as melhores práticas de proteção Serviços da AWS e mapeiam as diretrizes para controles de segurança em várias estruturas (incluindo o Instituto Nacional de Padrões e Tecnologia (NIST), o Conselho de Padrões de Segurança do Setor de Cartões de Pagamento (PCI) e a Organização Internacional de Padronização (ISO)).
- Avaliação de recursos com regras no Guia do AWS Config desenvolvedor — O AWS Config serviço avalia o quão bem suas configurações de recursos estão em conformidade com as práticas internas, as diretrizes e os regulamentos do setor.

- AWS Security Hub— Isso AWS service (Serviço da AWS) fornece uma visão abrangente do seu estado de segurança interno AWS. O Security Hub usa controles de segurança para avaliar os recursos da AWS e verificar a conformidade com os padrões e as práticas recomendadas do setor de segurança. Para obter uma lista dos serviços e controles aceitos, consulte a Referência de controles do Security Hub.
- Amazon GuardDuty — Isso AWS service (Serviço da AWS) detecta possíveis ameaças às suas cargas de trabalho Contas da AWS, contêineres e dados monitorando seu ambiente em busca de atividades suspeitas e maliciosas. GuardDuty pode ajudá-lo a atender a vários requisitos de conformidade, como o PCI DSS, atendendo aos requisitos de detecção de intrusões exigidos por determinadas estruturas de conformidade.
- AWS Audit Manager— Isso AWS service (Serviço da AWS) ajuda você a auditar continuamente seu AWS uso para simplificar a forma como você gerencia o risco e a conformidade com as regulamentações e os padrões do setor.

Resiliência no serviço de computação AWS paralela

A infraestrutura AWS global é construída em torno Regiões da AWS de zonas de disponibilidade. Regiões da AWS fornecem várias zonas de disponibilidade fisicamente separadas e isoladas, conectadas a redes de baixa latência, alta taxa de transferência e alta redundância. Com as zonas de disponibilidade, é possível projetar e operar aplicações e bancos de dados que automaticamente executam o failover entre as zonas sem interrupção. As zonas de disponibilidade são altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre zonas de disponibilidade Regiões da AWS e zonas de disponibilidade, consulte Infraestrutura AWS global.

Segurança de infraestrutura no serviço de computação AWS paralela

Como um serviço gerenciado, o AWS Parallel Computing Service é protegido pela segurança de rede AWS global. Para obter informações sobre serviços AWS de segurança e como AWS proteger a infraestrutura, consulte AWS Cloud Security. Para projetar seu AWS ambiente usando as melhores práticas de segurança de infraestrutura, consulte Proteção de infraestrutura no Security Pillar AWS Well-Architected Framework.

Você usa chamadas de API AWS publicadas para acessar o AWS PCS pela rede. Os clientes devem oferecer compatibilidade com:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Conjuntos de criptografia com perfect forward secrecy (PFS) como DHE (Ephemeral Diffie-Hellman) ou ECDHE (Ephemeral Elliptic Curve Diffie-Hellman). A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

Além disso, as solicitações devem ser assinadas usando um ID da chave de acesso e uma chave de acesso secreta associada a uma entidade principal do IAM. Ou é possível usar o AWS Security Token Service (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

Quando o AWS PCS cria um cluster, o serviço inicia o controlador Slurm em uma conta de propriedade do serviço, separada dos nós de computação em sua conta. Para unir a comunicação entre o controlador e os nós de computação, o AWS PCS cria uma interface de rede elástica (ENI) entre contas em sua VPC. O controlador Slurm usa o ENI para gerenciar e se comunicar com os nós de computação em diferentes Contas da AWS, mantendo a segurança e o isolamento dos recursos e, ao mesmo tempo, facilitando a eficiência da HPC e das operações. AI/ML

Análise e gerenciamento de vulnerabilidades no Serviço de Computação AWS Paralela

A configuração e os controles de TI são uma responsabilidade compartilhada entre você AWS e você. Para obter mais informações, consulte o modelo de responsabilidade AWS compartilhada. AWS lida com tarefas básicas de segurança para a infraestrutura subjacente na conta de serviço, como corrigir o sistema operacional nas instâncias do controlador, configuração do firewall e recuperação de desastres da AWS infraestrutura. Esses procedimentos foram revisados e certificados por terceiros certificados. Para obter mais detalhes, consulte Práticas recomendadas de segurança, identidade e conformidade.

> **Note**
>
> Os controladores Slurm não estão disponíveis enquanto os atualizamos. Os trabalhos em execução não são afetados. Os trabalhos enviados quando o controlador do cluster não está disponível são mantidos até que o controlador esteja disponível.

Você é responsável pela segurança da infraestrutura subjacente em seu Conta da AWS:

- Mantenha seu código, incluindo atualizações e patches de segurança.
- Corrija e atualize o sistema operacional na Amazon Machine Image (AMI) para seus grupos de nós de computação e atualize seus grupos de nós de computação para usar a AMI atualizada.
- Atualize o agendador para mantê-lo dentro das versões compatíveis. Atualize a AMI para seus grupos de nós de computação e atualize seu grupo de nós de computação para usar a AMI atualizada.
- Autentique e criptografe a comunicação entre clientes usuários e os nós aos quais eles se conectam.

Para obter mais informações sobre como atualizar a AMI para seus grupos de nós de computação, consulteAmazon Machine Images (AMIs) para AWS PCS.

Prevenção contra o ataque do “substituto confuso” em todos os serviços

O problema de adjunto confuso é um problema de segurança em que uma entidade que não tem permissão para executar uma ação pode coagir outra entidade mais privilegiada a executá-la. Em AWS, a falsificação de identidade entre serviços pode resultar no problema confuso do deputado. A personificação entre serviços pode ocorrer quando um serviço (o serviço de chamada) chama outro serviço (o serviço chamado). O serviço de chamada pode ser manipulado de modo a usar suas permissões para atuar nos recursos de outro cliente de uma forma na qual ele não deveria ter permissão para acessar. Para evitar isso, a AWS fornece ferramentas que ajudam você a proteger seus dados para todos os serviços com entidades principais de serviço que receberam acesso aos recursos em sua conta.

Recomendamos usar as `aws:SourceArn`chaves de contexto de condição `aws:SourceAccount`global nas políticas de recursos para limitar as permissões que o Serviço de Computação AWS Paralela (AWS PCS) concede a outro serviço ao recurso. Use `aws:SourceArn` se quiser que apenas um recurso seja associado ao acesso entre serviços. Use `aws:SourceAccount` se quiser permitir que qualquer recurso nessa conta seja associado ao uso entre serviços.

A maneira mais eficaz de se proteger contra o problema do substituto confuso é usar a chave de contexto de condição global `aws:SourceArn` com o ARN completo do recurso. Se você não souber

o ARN completo do recurso ou especificar vários recursos, use a chave de condição de contexto global `aws:SourceArn` com caracteres curinga (*) para as partes desconhecidas do ARN. Por exemplo, .`arn:aws:`*`servicename`*`:*:`*`123456789012`*`:*`

Se o valor de `aws:SourceArn` não contiver o ID da conta, como um ARN de bucket do Amazon S3, você deverá usar ambas as chaves de contexto de condição global para limitar as permissões.

O valor de `aws:SourceArn` deve ser um ARN de cluster.

O exemplo a seguir mostra como você pode usar as chaves de contexto de condição `aws:SourceAccount` global `aws:SourceArn` e as chaves de contexto no AWS PCS para evitar o confuso problema substituto.

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

Função do IAM para EC2 instâncias da Amazon provisionadas como parte de um grupo de nós de computação

AWS O PCS orquestra automaticamente a EC2 capacidade da Amazon para cada um dos grupos de nós de computação configurados em um cluster. Ao criar um grupo de nós de computação, os usuários devem fornecer um perfil de instância do IAM por meio do `iamInstanceProfileArn`

campo. O perfil da instância especifica as permissões associadas às instâncias provisionadas EC2 . AWS O PCS aceita qualquer função que tenha `AWSPCS` como prefixo do nome da função ou `/aws-pcs/` como parte do caminho da função. A `iam:PassRole` permissão é necessária na identidade do IAM (usuário ou função) que cria ou atualiza um grupo de nós de computação. Quando um usuário chama as ações `CreateComputeNodeGroup` ou a `UpdateComputeNodeGroup` API, o AWS PCS verifica se o usuário tem permissão para realizar a `iam:PassRole` ação.

O exemplo de política a seguir concede permissões para passar somente perfis do IAM cujo nome comece com `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"
                    ]
                }
            }
        }
    ]
}
```

Melhores práticas de segurança para serviços de computação AWS paralela

Esta seção descreve as melhores práticas de segurança específicas do Serviço de Computação AWS Paralela (AWS PCS). Para saber mais sobre as melhores práticas de segurança em AWS, consulte Melhores práticas de segurança, identidade e conformidade.

Segurança relacionada à AMI

- Não use a amostra AWS PCS AMIs para cargas de trabalho de produção. A amostra não AMIs tem suporte e é destinada apenas para testes.

- Atualize regularmente o sistema operacional e o software na AMI para seus grupos de nós de computação para reduzir as vulnerabilidades.
- Use somente pacotes AWS PCS oficiais autenticados baixados de AWS fontes oficiais.
- Atualize regularmente os pacotes AWS PCS na AMI para grupos de nós de computação e atualize os nós de computação para usar a AMI atualizada. Considere automatizar esse processo para minimizar as vulnerabilidades.

Para obter mais informações, consulte Imagens personalizadas da Amazon Machine (AMIs) para AWS PCS.

Segurança do Slurm Workload Manager

- Implemente controles de acesso e restrições de rede para proteger os nós de controle e computação do Slurm. Só permita que usuários e sistemas confiáveis enviem trabalhos e acessem os comandos de gerenciamento do Slurm.
- Use os recursos de segurança integrados do Slurm, como a autenticação do Slurm, para garantir que os envios de trabalhos e as comunicações sejam autenticados.
- Atualize as versões do Slurm para manter as operações e o suporte ao cluster sem problemas.

> **Important**
>
> Qualquer cluster que usa uma versão do Slurm que tenha atingido o fim da vida útil do suporte (EOSL) é interrompido imediatamente. Use o link na parte superior das páginas do guia do usuário para assinar o feed RSS da documentação do AWS PCS e receber uma notificação quando uma versão do Slurm se aproximar do EOSL.

Para obter mais informações, consulte Versões Slurm no PCS AWS.

Monitorar e registrar em log

- Use o Amazon CloudWatch Logs e AWS CloudTrail para monitorar e registrar ações em seus clusters Conta da AWS e. Use os dados para solução de problemas e auditoria.

Segurança de rede

- Implante seus clusters de AWS PCS em uma VPC separada para isolar seu ambiente de HPC de outros tráfegos de rede.
- Use grupos de segurança e listas de controle de acesso à rede (ACLs) para controlar o tráfego de entrada e saída para instâncias e sub-redes do AWS PCS.
- Use AWS PrivateLink nossos endpoints VPC para manter o tráfego de rede entre seus clusters e outros AWS serviços dentro da rede. AWS Para obter mais informações, consulte Acesso AWS Parallel Computing Service usando um endpoint de interface ()AWS PrivateLink.

Registro e monitoramento para AWS PCS

O monitoramento é uma parte importante da manutenção da confiabilidade, disponibilidade e desempenho do AWS PCS e de seus outros recursos da AWS. A AWS fornece as seguintes ferramentas de monitoramento para monitorar o AWS PCS, relatar quando algo está errado e realizar ações automáticas quando apropriado:

- A Amazon CloudWatch monitora seus AWS recursos e os aplicativos em que você executa AWS em tempo real. Você pode coletar e rastrear métricas, criar painéis personalizados e definir alarmes que o notificam ou que realizam ações quando uma métrica especificada atinge um limite definido. Por exemplo, você pode CloudWatch rastrear o uso da CPU ou outras métricas de suas EC2 instâncias da Amazon e iniciar automaticamente novas instâncias quando necessário. Para obter mais informações, consulte o Guia CloudWatch do usuário da Amazon.
- O Amazon CloudWatch Logs permite que você monitore, armazene e acesse seus arquivos de log de EC2 instâncias da Amazon e de outras fontes. CloudTrail CloudWatch Os registros podem monitorar as informações nos arquivos de log e notificá-lo quando determinados limites forem atingidos. É possível também arquivar seus dados de log em armazenamento resiliente. Para obter mais informações, consulte o Guia do usuário do Amazon CloudWatch Logs.
- AWS CloudTrailcaptura chamadas de API e eventos relacionados feitos por ou em nome de sua AWS conta e entrega os arquivos de log para um bucket do Amazon S3 que você especificar. Você pode identificar quais usuários e contas ligaram AWS, o endereço IP de origem a partir do qual as chamadas foram feitas e quando elas ocorreram. Para obter mais informações, consulte o Guia do usuário do AWS CloudTrail.

Registros de conclusão de trabalhos no AWS PCS

Os registros de conclusão do trabalho fornecem detalhes importantes sobre seus trabalhos do Serviço de Computação AWS Paralela (AWS PCS) quando eles são concluídos, sem custo adicional. Você pode usar outros AWS serviços para acessar e processar seus dados de log, como Amazon CloudWatch Logs, Amazon Simple Storage Service (Amazon S3) e Amazon Data Firehose AWS ; o PCS registra metadados sobre seus trabalhos, como os seguintes.

- ID e nome do Job
- Informações do usuário e do grupo
- Estado do trabalho (como`COMPLETED`,`FAILED`,`CANCELLED`)

- Partição usada
- Limites de tempo
- Horários de início, término, envio e qualificáveis
- Lista e contagem de nós
- Contagem de processadores
- Diretório de trabalho
- Uso de recursos (CPU, memória)
- Códigos de saída
- Detalhes do nó (nomes, instância IDs, tipos de instância)

Sumário

Pré-requisitos

O diretor do IAM que gerencia o cluster AWS PCS deve permitir a `pcs:AllowVendedLogDeliveryForResource` ação.

O exemplo a seguir da política do IAM concede as permissões necessárias.

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "PcsAllowVendedLogsDelivery",
         "Effect": "Allow",
         "Action": ["pcs:AllowVendedLogDeliveryForResource"],
         "Resource": [
```

```
            "arn:aws:pcs:::cluster/*"
         ]
      }
   ]
}
```

Configurar registros de conclusão do trabalho

Você pode configurar registros de conclusão de tarefas para seu cluster AWS PCS com o AWS Management Console ou AWS CLI.

AWS Management Console

Para configurar registros de conclusão de trabalhos com o console

1. Abra o console AWS PCS.
2. No painel de navegação, escolha Clusters.
3. Escolha o cluster ao qual você deseja adicionar os registros de conclusão do trabalho.
4. Na página de detalhes do cluster, escolha a guia Registros.
5. Em Job Conclution Logs, escolha Add para adicionar até 3 destinos de entrega de CloudWatch logs entre Logs, Amazon S3 e Firehose.
6. Escolha Atualizar entregas de registros.

AWS CLI

Para configurar registros de conclusão do trabalho com o AWS CLI

1. Crie um destino de entrega de registros:

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

Substitua:

- *region*— O Região da AWS local onde você deseja criar o destino, como `us-east-1`
- *pcs-logs-destination*— Um nome para o destino

- *resource-arn*— O Amazon Resource Name (ARN) de um grupo de CloudWatch logs do Logs, bucket S3 ou stream de entrega do Firehose.

Para obter mais informações, consulte PutDeliveryDestinationa Referência da API Amazon CloudWatch Logs.

2. Defina o cluster PCS como uma fonte de entrega de registros:

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

Substitua:

- *region*— O Região da AWS do seu cluster, como `us-east-1`
- *cluster-logs-source-name*— Um nome para a fonte
- *cluster-arn*— o ARN do seu AWS cluster PCS

Para obter mais informações, consulte PutDeliverySourcea Referência da API Amazon CloudWatch Logs.

3. Conecte a fonte de entrega ao destino da entrega:

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

Substitua:

- *region*— O Região da AWS, como `us-east-1`
- *cluster-logs-source*— O nome da sua fonte de entrega
- *destination-arn*— O ARN do seu destino de entrega

Para obter mais informações, consulte CreateDeliverya Referência da API Amazon CloudWatch Logs.

Como encontrar registros de conclusão de trabalhos

Você pode configurar destinos de log no CloudWatch Logs e no Amazon S3. AWS O PCS usa os seguintes nomes de caminhos estruturados e nomes de arquivos.

CloudWatch Registros

AWS O PCS usa o seguinte formato de nome para o stream de CloudWatch registros:

```
AWSLogs/PCS/cluster-id/jobcomp.log
```

Por exemplo: `AWSLogs/PCS/pcs_abc123de45/jobcomp.log`

Amazon S3

AWS O PCS usa o seguinte formato de nome para o caminho do S3:

```
AWSLogs/account-id/PCS/region/cluster-id/jobcomp/year/month/day/hour/
```

Por exemplo: `AWSLogs/111122223333/PCS/us-east-1/pcs_abc123de45/jobcomp/2025/06/19/11/`

AWS O PCS usa o seguinte formato de nome para os arquivos de log:

```
PCS_jobcomp_year-month-day-hour_cluster-id_random-id.log.gz
```

Por exemplo: `PCS_jobcomp_2025-06-19-11_pcs_abc123de45_04be080b.log.gz`

Campos do registro de conclusão do trabalho

AWS O PCS grava dados de registro de conclusão do trabalho como objetos JSON. O contêiner JSON `jobcomp` contém os detalhes do trabalho. A tabela a seguir descreve os campos dentro do `jobcomp` contêiner. Alguns campos só estão presentes em circunstâncias específicas, como para trabalhos de matriz ou trabalhos heterogêneos.

Campos do registro de conclusão do trabalho

Name	Valor de exemplo	Obrigatório	Observações
`job_id`	11	sim	Sempre presente com valor

Name	Valor de exemplo	Obrigatório	Observações
`user`	`"root"`	sim	Sempre presente com valor
`user_id`	`0`	sim	Sempre presente com valor
`group`	`"root"`	sim	Sempre presente com valor
`group_id`	`0`	sim	Sempre presente com valor
`name`	`"wrap"`	sim	Sempre presente com valor
`job_state`	`"COMPLETED"`	sim	Sempre presente com valor
`partition`	`"Hydra-MpiQueue-abcdef01-7"`	sim	Sempre presente com valor
`time_limit`	`"UNLIMITED"`	sim	Sempre presente, mas pode estar `"UNLIMITED"`
`start_time`	`"2025-06-19T10:58:57"`	sim	Sempre presente, mas pode estar `"Unknown"`
`end_time`	`"2025-06-19T10:58:57"`	sim	Sempre presente, mas pode estar `"Unknown"`
`node_list`	`"Hydra-MpiNG-abcdef01-2345-1"`	sim	Sempre presente com valor
`node_cnt`	`1`	sim	Sempre presente com valor
`proc_cnt`	`1`	sim	Sempre presente com valor

Name	Valor de exemplo	Obrigatório	Observações
work_dir	"/root"	sim	Sempre presente, mas pode estar "Unknown"
reservation_name	"weekly_maintenance"	sim	Sempre presente, mas pode ser uma string vazia ""
tres.cpu	1	sim	Sempre presente com valor
tres.mem.val	600	sim	Sempre presente com valor
tres.mem.unit	"M"	sim	Pode ser "M" ou "bb"
tres.node	1	sim	Sempre presente com valor
tres.billing	1	sim	Sempre presente com valor
account	"finance"	sim	Sempre presente, mas pode ser uma string vazia ""
qos	"normal"	sim	Sempre presente, mas pode ser uma string vazia ""
wc_key	"project_1"	sim	Sempre presente, mas pode ser uma string vazia ""
cluster	"unknown"	sim	Sempre presente, mas pode estar "unknown"
submit_time	"2025-06-19T10:55:46"	sim	Sempre presente, mas pode estar "Unknown"

Name	Valor de exemplo	Obrigatório	Observações
`eligible_time`	`"2025-06-19T10:55:46"`	sim	Sempre presente, mas pode estar `"Unknown"`
`array_job_id`	`12`	não	Presente somente se o trabalho for um trabalho de matriz
`array_task_id`	`1`	não	Presente somente se o trabalho for um trabalho de matriz
`het_job_id`	`10`	não	Presente apenas se o trabalho for heterogêneo
`het_job_offset`	`0`	não	Presente apenas se o trabalho for heterogêneo
`derived_exit_code_status`	`0`	sim	Sempre presente com valor
`derived_exit_code_signal`	`0`	sim	Sempre presente com valor
`exit_code_status`	`0`	sim	Sempre presente com valor
`exit_code_signal`	`0`	sim	Sempre presente com valor
`node_details[0].name`	`"Hydra-MpiNG-abcdef01-2345-1"`	não	Sempre presente, mas `node_details` pode estar `"[]"`

Name	Valor de exemplo	Obrigatório	Observações
`node_details[0].instance_id`	`"i-0abcdef01234567a"`	não	Sempre presente, mas `node_details` pode estar `"[]"`
`node_details[0].instance_type`	`"t4g.micro"`	não	Sempre presente, mas `node_details` pode estar `"[]"`

Exemplos de registros de conclusão de trabalhos

Os exemplos a seguir mostram registros de conclusão de trabalhos para vários tipos e estados de trabalhos:


```
{ "jobcomp": { "job_id": 1, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T16:32:57",
 "end_time": "2025-06-19T16:33:03", "node_list": "Hydra-MpiNG-abcdef01-2345-
[1-2]", "node_cnt": 2, "proc_cnt": 2, "work_dir": "/usr/bin", "reservation_name":
 "", "tres": { "cpu": 2, "mem": { "val": 1944, "unit": "M" }, "node": 2,
 "billing": 2 }, "account": "", "qos": "", "wc_key": "", "cluster": "unknown",
 "submit_time": "2025-06-19T16:29:40", "eligible_time": "2025-06-19T16:29:41",
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc123def45678", "instance_type": "t4g.micro" }, { "name":
 "Hydra-MpiNG-abcdef01-2345-2", "instance_id": "i-0def456abc78901", "instance_type":
 "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 2, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T16:33:13",
 "end_time": "2025-06-19T16:33:14", "node_list": "Hydra-MpiNG-abcdef01-2345-
[1-2]", "node_cnt": 2, "proc_cnt": 2, "work_dir": "/usr/bin", "reservation_name":
 "", "tres": { "cpu": 2, "mem": { "val": 1944, "unit": "M" }, "node": 2,
 "billing": 2 }, "account": "", "qos": "", "wc_key": "", "cluster": "unknown",
 "submit_time": "2025-06-19T16:33:13", "eligible_time": "2025-06-19T16:33:13",
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
```

```
 "instance_id": "i-0abc123def45678", "instance_type": "t4g.micro" }, { "name":
 "Hydra-MpiNG-abcdef01-2345-2", "instance_id": "i-0def456abc78901", "instance_type":
 "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 3, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T22:58:57", "end_time":
 "2025-06-19T22:58:57", "node_list": "Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 972, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T22:55:46",
 "eligible_time": "2025-06-19T22:55:46", "derived_exit_code_status": 0,
 "derived_exit_code_signal": 0, "exit_code_status": 0, "exit_code_signal":
 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1", "instance_id":
 "i-0abc234def56789", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 4, "user": "root", "user_id": 0, "group": "root",
 "group_id": 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-
MpiQueue-abcdef01-7", "time_limit": "525600", "start_time": "2025-06-19T23:04:27",
 "end_time": "2025-06-19T23:04:27", "node_list": "Hydra-MpiNG-abcdef01-2345-
[1-2]", "node_cnt": 2, "proc_cnt": 2, "work_dir": "/root", "reservation_name":
 "", "tres": { "cpu": 2, "mem": { "val": 1944, "unit": "M" }, "node": 2,
 "billing": 2 }, "account": "", "qos": "", "wc_key": "", "cluster": "unknown",
 "submit_time": "2025-06-19T23:01:38", "eligible_time": "2025-06-19T23:01:38",
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc234def56789", "instance_type": "t4g.micro" }, { "name":
 "Hydra-MpiNG-abcdef01-2345-2", "instance_id": "i-0def345abc67890", "instance_type":
 "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 5, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "FAILED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:09:00", "end_time":
 "2025-06-19T23:09:00", "node_list": "(null)", "node_cnt": 0, "proc_cnt": 0,
 "work_dir": "/root", "reservation_name": "", "tres": { "cpu": 1, "mem": { "val":
 1, "unit": "G" }, "node": 1, "billing": 1 }, "account": "", "qos": "", "wc_key":
 "", "cluster": "unknown", "submit_time": "2025-06-19T23:09:00", "eligible_time":
 "2025-06-19T23:09:00", "derived_exit_code_status": 0, "derived_exit_code_signal": 0,
 "exit_code_status": 0, "exit_code_signal": 1, "node_details": [] } }
{ "jobcomp": { "job_id": 6, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "CANCELLED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:09:36",
 "end_time": "2025-06-19T23:09:36", "node_list": "(null)", "node_cnt": 0, "proc_cnt":
 0, "work_dir": "/root", "reservation_name": "", "tres": { "cpu": 1, "mem":
 { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "", "qos":
 "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:09:35",
 "eligible_time": "2025-06-19T23:09:36", "het_job_id": 6, "het_job_offset": 0,
```

```
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status": 0,
 "exit_code_signal": 1, "node_details": [] } }
{ "jobcomp": { "job_id": 7, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "CANCELLED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:10:03",
 "end_time": "2025-06-19T23:10:03", "node_list": "(null)", "node_cnt": 0, "proc_cnt":
 0, "work_dir": "/root", "reservation_name": "", "tres": { "cpu": 1, "mem":
 { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "", "qos":
 "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:10:03",
 "eligible_time": "2025-06-19T23:10:03", "het_job_id": 7, "het_job_offset": 0,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status": 0,
 "exit_code_signal": 1, "node_details": [] } }
{ "jobcomp": { "job_id": 8, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:11:24", "end_time":
 "2025-06-19T23:11:24", "node_list": "Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:11:23",
 "eligible_time": "2025-06-19T23:11:23", "het_job_id": 8, "het_job_offset": 0,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc234def56789", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 9, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:11:24", "end_time":
 "2025-06-19T23:11:24", "node_list": "Hydra-MpiNG-abcdef01-2345-2", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:11:23",
 "eligible_time": "2025-06-19T23:11:23", "het_job_id": 8, "het_job_offset": 1,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-2",
 "instance_id": "i-0def345abc67890", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 10, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:12:24", "end_time":
 "2025-06-19T23:12:24", "node_list":"Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:12:14",
 "eligible_time": "2025-06-19T23:12:14", "het_job_id": 10, "het_job_offset": 0,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
```

```
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc234def56789", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 11, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:12:24", "end_time":
 "2025-06-19T23:12:24", "node_list":"Hydra-MpiNG-abcdef01-2345-2", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 600, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:12:14",
 "eligible_time": "2025-06-19T23:12:14", "het_job_id": 10, "het_job_offset": 1,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-2",
 "instance_id": "i-0def345abc67890", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 13, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:47:57", "end_time":
 "2025-06-19T23:47:58", "node_list":"Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 972, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:43:56",
 "eligible_time": "2025-06-19T23:43:56" , "array_job_id": 12, "array_task_id": 1,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc345def67890", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 12, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:47:58", "end_time":
 "2025-06-19T23:47:58", "node_list":"Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 972, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:43:56",
 "eligible_time": "2025-06-19T23:43:56" , "array_job_id": 12, "array_task_id": 2,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc345def67890", "instance_type": "t4g.micro" } ] } }
```


Logs do agendador no AWS PCS

Você pode configurar o AWS PCS para enviar dados de registro detalhados do seu agendador de cluster para o Amazon CloudWatch Logs, o Amazon Simple Storage Service (Amazon S3) e o Amazon Data Firehose. Isso pode ajudar no monitoramento e na solução de problemas.

Sumário

Pré-requisitos

O diretor do IAM que gerencia o cluster AWS PCS deve permitir a `pcs:AllowVendedLogDeliveryForResource` ação.

O exemplo a seguir da política do IAM concede as permissões necessárias.

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

Configurar registros do agendador

Você pode configurar os registros do agendador para seu cluster AWS PCS com o AWS Management Console ou AWS CLI.

AWS Management Console

Para configurar os registros do agendador com o console

1. Abra o console AWS PCS.
2. No painel de navegação, escolha Clusters.
3. Escolha o cluster ao qual você deseja adicionar os registros do agendador.

4. Na página de detalhes do cluster, escolha a guia Registros.
5. Em Scheduler Logs, escolha Add para adicionar até 3 destinos de entrega de CloudWatch logs entre Logs, Amazon S3 e Firehose.
6. Escolha Atualizar entregas de registros.

AWS CLI

Para configurar os registros do agendador com o AWS CLI

1. Crie um destino de entrega de registros:

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

Substitua:

- *region*— O Região da AWS local onde você deseja criar o destino, como `us-east-1`
- *pcs-logs-destination*— Um nome para o destino
- *resource-arn*— O Amazon Resource Name (ARN) de um grupo de CloudWatch logs do Logs, bucket do S3 ou stream de entrega do Firehose.

Para obter mais informações, consulte PutDeliveryDestinationa Referência da API Amazon CloudWatch Logs.

2. Defina o cluster PCS como uma fonte de entrega de registros:

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

Substitua:

- *region*— O Região da AWS do seu cluster, como `us-east-1`
- *cluster-logs-source-name*— Um nome para a fonte
- *cluster-arn*— o ARN do seu AWS cluster PCS

Para obter mais informações, consulte PutDeliverySourcea Referência da API Amazon CloudWatch Logs.

3. Conecte a fonte de entrega ao destino da entrega:

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

Substitua:

- *region*— O Região da AWS, como `us-east-1`
- *cluster-logs-source*— O nome da sua fonte de entrega
- *destination-arn*— O ARN do seu destino de entrega

Para obter mais informações, consulte CreateDeliverya Referência da API Amazon CloudWatch Logs.

Caminhos e nomes do fluxo de registros do agendador

O caminho e o nome dos registros do agendador AWS PCS dependem do tipo de destino.

- CloudWatch Logs
  - Um stream de CloudWatch registros segue essa convenção de nomenclatura.

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

Example

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

- S3 bucket
  - Um caminho de saída do bucket S3 segue esta convenção de nomenclatura:

```
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.
```

- Um nome de objeto S3 segue esta convenção:

```
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
```

Exemplo de registro de log do agendador

AWS Os registros do agendador PCS são estruturados. Eles incluem campos como identificador do cluster, tipo de agendador, versões principais e de patch, além da mensagem de log emitida pelo processo do controlador Slurm. Aqui está um exemplo.

```
{
    "resource_id": "s3431v9rx2",
    "resource_type": "PCS_CLUSTER",
    "event_timestamp": 1721230979,
    "log_level": "info",
    "log_name": "slurmctld",
    "scheduler_type": "slurm",
    "scheduler_major_version": "24.11",
    "scheduler_patch_version": "5",
    "node_type": "controller_primary",
    "message": "[2024-07-17T15:42:58.614+00:00] Running as primary controller\n"
}
```

Serviço de monitoramento de computação AWS paralela com a Amazon CloudWatch

CloudWatch A Amazon fornece monitoramento da integridade e do desempenho do seu cluster do AWS Parallel Computing Service (AWS PCS) coletando métricas do cluster em intervalos. Essas

métricas são mantidas, permitindo que você acesse dados históricos e obtenha insights sobre o desempenho do seu cluster ao longo do tempo.

CloudWatch também permite monitorar as EC2 instâncias lançadas pelo AWS PCS para atender aos seus requisitos de escalabilidade. Embora você possa inspecionar registros em instâncias em execução, CloudWatch as métricas e os dados de registro geralmente são excluídos quando as instâncias são encerradas. No entanto, você pode configurar o CloudWatch agente em instâncias usando um modelo de EC2 lançamento para manter métricas e registros mesmo após o encerramento da instância, permitindo monitoramento e análise de longo prazo.

Explore os tópicos desta seção para saber mais sobre como monitorar o uso do AWS PCS CloudWatch.

Tópicos

Monitorando métricas do AWS PCS usando CloudWatch

Você pode monitorar a integridade do cluster AWS PCS usando a Amazon CloudWatch, que coleta dados do seu cluster e os transforma em métricas quase em tempo real. Essas estatísticas são mantidas por um período de 15 meses, para que você possa acessar informações históricas e ter uma perspectiva melhor sobre o desempenho do seu cluster. As métricas do cluster são enviadas CloudWatch em períodos de 1 minuto. Para obter mais informações sobre CloudWatch, consulte O que é a Amazon CloudWatch? no Guia do CloudWatch usuário da Amazon.

AWS O PCS publica as seguintes métricas no namespace AWS/PCS em. CloudWatch Eles têm uma única dimensão,`ClusterId`.

Nome	Descrição	Unidades
ActualCapacity	IdleCapacity + UtilizedCapacity	Contagem
CapacityUtilization	UtilizedCapacity / ActualCapacity	Contagem

Nome	Descrição	Unidades
DesiredCapacity	ActualCapacity + PendingCapacity	Contagem
IdleCapacity	Contagem de instâncias em execução, mas não alocadas para trabalhos	Contagem
UtilizedCapacity	Contagem de instâncias em execução e alocadas para trabalhos	Contagem

Monitoramento de instâncias AWS PCS usando a Amazon CloudWatch

O AWS PCS lança EC2 instâncias da Amazon conforme necessário para atender aos requisitos de escalabilidade definidos em seus grupos de nós de computação do PCS. Você pode monitorar essas instâncias enquanto elas estão em execução usando a Amazon CloudWatch. Você pode inspecionar os registros das instâncias em execução fazendo login nelas e usando ferramentas de linha de comando interativas. No entanto, por padrão, os dados de CloudWatch métricas só são retidos por um período limitado quando uma instância é encerrada, e os registros da instância geralmente são excluídos junto com os volumes do EBS que apoiam a instância. Para reter métricas ou dados de registro das instâncias lançadas pelo PCS após o encerramento, você pode configurar o CloudWatch agente em suas instâncias com um modelo de EC2 execução. Este tópico fornece uma visão geral do monitoramento de instâncias em execução e fornece exemplos de como configurar métricas e registros de instâncias persistentes.

Monitorando instâncias em execução

Encontrando instâncias do AWS PCS

Para monitorar instâncias lançadas pelo PCS, encontre as instâncias em execução associadas a um cluster ou grupo de nós de computação. Em seguida, no EC2 console de uma determinada instância, inspecione as seções Status e alarmes e Monitoramento. Se o acesso de login estiver configurado para essas instâncias, você poderá se conectar a elas e inspecionar vários arquivos de log nas instâncias. Para obter mais informações sobre como identificar quais instâncias são gerenciadas pelo PCS, consulteEncontrando instâncias de grupos de nós de computação no AWS PCS.

Habilitando métricas detalhadas

Por padrão, as métricas da instância são coletadas em intervalos de 5 minutos. Para coletar métricas em intervalos de um minuto, ative o CloudWatch monitoramento detalhado em seu modelo de lançamento do grupo de nós de computação. Para obter mais informações, consulte Ativar o CloudWatch monitoramento detalhado.

Configurando métricas e registros de instâncias persistentes

Você pode reter as métricas e os registros de suas instâncias instalando e configurando o CloudWatch agente da Amazon nelas. Isso consiste em três etapas principais:

1. Crie uma configuração de CloudWatch agente.
2. Armazene a configuração onde ela possa ser recuperada pelas instâncias do PCS.
3. Escreva um modelo de EC2 lançamento que instale o software do CloudWatch agente, busque sua configuração e inicie o CloudWatch agente usando a configuração.

Para obter mais informações, consulte Coletar métricas, registros e rastreamentos com o CloudWatch agente no Guia CloudWatch do usuário da Amazon Usando modelos de EC2 lançamento da Amazon com AWS PCS e.

Criar uma configuração de CloudWatch agente

Antes de implantar o CloudWatch agente em suas instâncias, você deve gerar um arquivo de configuração JSON que especifique as métricas, os registros e os rastreamentos a serem coletados. Os arquivos de configuração podem ser criados usando um assistente ou manualmente, usando um editor de texto. O arquivo de configuração será criado manualmente para esta demonstração.

Em um computador em que você tenha a AWS CLI instalada, crie um arquivo de CloudWatch configuração chamado config.json com o conteúdo a seguir. Você também pode usar o seguinte URL para baixar uma cópia do arquivo.

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

Observações

- Os caminhos de log no arquivo de amostra são para o Amazon Linux 2. Se suas instâncias usarem um sistema operacional básico diferente, altere os caminhos conforme apropriado.
- Para capturar outros registros, adicione outras entradas abaixo`collect_list`.

- Os valores em `{brackets}` são variáveis modeladas. Para obter a lista completa das variáveis suportadas, consulte Criar ou editar manualmente o arquivo de configuração do CloudWatch agente no Guia CloudWatch do usuário da Amazon.
- Você pode optar por omitir `logs` ou `metrics` se não quiser coletar esses tipos de informações.

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

Esse arquivo instrui o CloudWatch agente a monitorar vários arquivos que podem ser úteis no diagnóstico de erros na inicialização, autenticação e login da instância e em outros domínios de solução de problemas. Isso inclui:

- `/var/log/cloud-init.log`— Saída do estágio inicial da configuração da instância
- `/var/log/cloud-init-output.log`— Saída de comandos que são executados durante a configuração da instância

- `/var/log/amazon/pcs/bootstrap.log`— Saída de operações específicas do PC que são executadas durante a configuração da instância
- `/var/log/slurmd.log`— Saída do daemon slurmd do gerenciador de carga de trabalho Slurm
- `/var/log/messages`— Mensagens do sistema do kernel, serviços do sistema e aplicativos
- `/var/log/secure`— Registros relacionados a tentativas de autenticação, como SSH, sudo e outros eventos de segurança

Os arquivos de log são enviados para um grupo de CloudWatch log chamado`/PCSLogs/instances`. Os fluxos de log são uma combinação do ID da instância e do nome base do arquivo de log. O grupo de registros tem um tempo de retenção de 30 dias.

Além disso, o arquivo instrui o CloudWatch agente a coletar várias métricas comuns, agregando-as por ID da instância.

Armazene a configuração

O arquivo de configuração do CloudWatch agente precisa ser armazenado onde possa ser acessado pelas instâncias do nó de computação do PCS. Há duas maneiras comuns de fazer isso. Você pode carregá-lo em um bucket do Amazon S3 ao qual suas instâncias do grupo de nós computacionais terão acesso por meio de seu perfil de instância. Como alternativa, você pode armazená-lo como um parâmetro SSM no Amazon Systems Manager Parameter Store.

Fazer upload para um bucket do S3

Para armazenar seu arquivo no S3, use os comandos da AWS CLI a seguir. Antes de executar o comando, faça estas substituições:

- *`amzn-s3-demo-bucket`*Substitua pelo seu próprio nome de bucket do S3

Primeiro, (isso é opcional se você tiver um bucket existente), crie um bucket para armazenar seus arquivos de configuração.

```
aws s3 mb s3://amzn-s3-demo-bucket
```

Em seguida, faça o upload do arquivo para o bucket.

```
aws s3 cp ./config.json s3://amzn-s3-demo-bucket/
```

Armazenar como um parâmetro SSM

Para armazenar seu arquivo como um parâmetro SSM, use o comando a seguir. Antes de executar o comando, faça estas substituições:

- *region-code*Substitua pela região da AWS em que você está trabalhando com o AWS PCS.
- (Opcional) *AmazonCloudWatch-PCS* Substitua o parâmetro pelo seu próprio nome. Observe que, se você alterar o prefixo do nome de, `AmazonCloudWatch-` precisará adicionar especificamente o acesso de leitura ao parâmetro SSM no perfil da instância do seu grupo de nós.

```
aws ssm put-parameter \
   --region region-code \
   --name "AmazonCloudWatch-PCS" \
   --type String \
   --value file://config.json
```

Escreva um modelo de EC2 lançamento

Os detalhes específicos do modelo de lançamento dependem de seu arquivo de configuração estar armazenado no S3 ou no SSM.

Use uma configuração armazenada no S3

Esse script instala o CloudWatch agente, importa um arquivo de configuração de um bucket do S3 e inicia o CloudWatch agente com ele. Substitua os seguintes valores nesse script pelos seus próprios detalhes:

- *amzn-s3-demo-bucket*— O nome de um bucket do S3 que sua conta pode ler
- */config.json*— Caminho relativo à raiz do bucket do S3 em que a configuração está armazenada

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

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

packages:
- amazon-cloudwatch-agent
```

```
runcmd:
- aws s3 cp s3://amzn-s3-demo-bucket/config.json /etc/s3-cw-config.json
- /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m
 ec2 -s -c file://etc/s3-cw-config.json

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

O perfil da instância do IAM para o grupo de nós deve ter acesso ao bucket. Aqui está um exemplo de política do IAM para o bucket no script de dados do usuário acima.

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

Observe também que as instâncias devem permitir tráfego de saída para o S3 e CloudWatch os endpoints. Isso pode ser feito usando grupos de segurança ou VPC endpoints, dependendo da arquitetura do cluster.

Use uma configuração armazenada no SSM

Esse script instala o CloudWatch agente, importa um arquivo de configuração de um parâmetro SSM e inicia o CloudWatch agente com ele. Substitua os seguintes valores nesse script pelos seus próprios detalhes:

- (Opcional) *AmazonCloudWatch-PCS* Substitua o parâmetro pelo seu próprio nome.

```
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==--
```

A política de instância do IAM para o grupo de nós deve ter o CloudWatchAgentServerPolicyanexado a ela.

Se o nome do seu parâmetro não começar com, `AmazonCloudWatch-` você precisará adicionar especificamente o acesso de leitura ao parâmetro SSM em seu perfil de instância de grupo de nós. Aqui está um exemplo de política do IAM que ilustra isso para prefixo*DOC-EXAMPLE-PREFIX*.

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "CustomCwSsmMParamReadOnly",
      "Effect" : "Allow",
      "Action" : [
        "ssm:GetParameter"
      ],
      "Resource" : "arn:aws:ssm:*:*:parameter/DOC-EXAMPLE-PREFIX*"
    }
  ]
}
```

Observe também que as instâncias devem permitir tráfego de saída para o SSM e CloudWatch os endpoints. Isso pode ser feito usando grupos de segurança ou VPC endpoints, dependendo da arquitetura do cluster.

Registrando chamadas de API do serviço de computação AWS paralela usando AWS CloudTrail

AWS O PCS é integrado com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço no AWS PCS. CloudTrail captura todas as chamadas de API para AWS PCS como eventos. As chamadas capturadas incluem chamadas do console do AWS PCS e chamadas de código para as operações da API do AWS PCS. Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para AWS PCS. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos. Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita ao AWS PCS, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Para saber mais sobre isso CloudTrail, consulte o Guia AWS CloudTrail do usuário.

AWS Informações do PCS em CloudTrail

CloudTrail é ativado no seu Conta da AWS quando você cria a conta. Quando a atividade ocorre no AWS PCS, essa atividade é registrada em um CloudTrail evento junto com outros eventos AWS de serviço no histórico de eventos. Você pode visualizar, pesquisar e baixar eventos recentes no seu Conta da AWS. Para obter mais informações, consulte Visualização de eventos com histórico de CloudTrail eventos.

Para um registro contínuo dos eventos em seu Conta da AWS, incluindo eventos para AWS PCS, crie uma trilha. Uma trilha permite CloudTrail entregar arquivos de log para um bucket do Amazon S3. Por padrão, quando você cria uma trilha no console, ela é aplicada a todas as Regiões da AWS. A trilha registra eventos de todas as regiões na AWS partição e entrega os arquivos de log ao bucket do Amazon S3 que você especificar. Além disso, você pode configurar outros AWS serviços para analisar e agir com base nos dados de eventos coletados nos CloudTrail registros. Para obter mais informações, consulte:

- Visão geral da criação de uma trilha
- CloudTrail serviços e integrações suportados
- Configurando notificações do Amazon SNS para CloudTrail
- Recebendo arquivos de CloudTrail log de várias regiões e Recebendo arquivos de CloudTrail log de várias contas

Todas as ações do AWS PCS são registradas CloudTrail e documentadas na Referência da API do Serviço de Computação AWS Paralela. Por exemplo, chamadas para as `DeleteCluster` ações `CreateComputeNodeGroupUpdateQueue`, e geram entradas nos arquivos de CloudTrail log.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar o seguinte:

- Se a solicitação foi feita com credenciais de usuário root ou AWS Identity and Access Management (IAM).
- Se a solicitação foi feita com credenciais de segurança temporárias de uma função ou de um usuário federado.
- Se a solicitação foi feita por outro AWS serviço.

Para obter mais informações, consulte Elemento userIdentity do CloudTrail .

Compreendendo as entradas do arquivo de CloudTrail log do AWS PCS

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do S3 que você especificar. CloudTrail os arquivos de log contêm uma ou mais entradas de log. Um evento representa uma única solicitação de qualquer fonte e inclui informações sobre a ação solicitada, a data e a hora da ação, os parâmetros da solicitação e assim por diante. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das chamadas públicas de API, portanto, eles não aparecem em nenhuma ordem específica.

O exemplo a seguir mostra uma entrada de CloudTrail registro para uma `CreateQueue` ação.

```
{
    "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"
}
```

Endpoints e cotas de serviço para PCS AWS

As seções a seguir descrevem os endpoints e as cotas de serviço do Serviço de Computação AWS Paralela (AWS PCS). As cotas de serviço, anteriormente chamadas de limites, são o número máximo de recursos ou operações de serviço para você. Conta da AWS

Você Conta da AWS tem cotas padrão para cada AWS serviço. A menos que especificado de outra forma, cada cota é específica da região . Você pode solicitar o aumento de algumas cotas, porém, algumas delas não podem ser aumentadas.

Para obter mais informações, consulte Service Quotas da AWS, na Referência geral da AWS .

Sumário

Service endpoints

Nome da região	Região	Endpoint	Protocolo
Leste dos EUA (Ohio)	us-east-2	pcs.us-east-2.amazonaws.com pcs-fips.us-east-2.amazonaws.com pcs-fips.us-east-2.api.aws pcs.us-east-2.api.aws	HTTPS
Leste dos EUA (Norte da Virgínia)	us-east-1	pcs.us-east-1.amazonaws.com	HTTPS

Nome da região	Região	Endpoint	Protocolo
		pcs-fips.us-east-1.amazonaws.com pcs-fips.us-east-1.api.aws pcs.us-east-1.api.aws	
Oeste dos EUA (Oregon)	us-west-2	pcs.us-west-2.amazonaws.com pcs-fips.us-west-2.amazonaws.com pcs-fips.us-west-2.api.aws pcs.us-west-2.api.aws	HTTPS
Ásia-Pacífico (Singapura)	ap-southeast-1	pcs.ap-southeast-1.amazonaws.com pcs.ap-southeast-1.api.aws	HTTPS
Ásia-Pacífico (Sydney)	ap-southeast-2	pcs.ap-southeast-2.amazonaws.com pcs.ap-southeast-2.api.aws	HTTPS
Ásia-Pacífico (Tóquio)	ap-northeast-1	pcs.ap-northeast-1.amazonaws.com pcs.ap-northeast-1.api.aws	HTTPS

Nome da região	Região	Endpoint	Protocolo
Europa (Frankfurt)	eu-central-1	pcs.eu-central-1.amazonaws.com pcs.eu-central-1.api.aws	HTTPS
Europa (Irlanda)	eu-west-1	pcs.eu-west-1.amazonaws.com pcs.eu-west-1.api.aws	HTTPS
Europa (Londres)	eu-west-2	pcs.eu-west-2.amazonaws.com pcs.eu-west-2.api.aws	HTTPS
Europa (Estocolmo)	eu-north-1	pcs.eu-north-1.amazonaws.com pcs.eu-north-1.api.aws	HTTPS
AWS GovCloud (Leste dos EUA)	us-gov-east-1	peças. us-gov-east-1.amazonaws.com dicas de peças. us-gov-east-1.amazonaws.com dicas de peças. us-gov-east-1.api.aws peças. us-gov-east-1.api.aws	HTTPS

Nome da região	Região	Endpoint	Protocolo
AWS GovCloud (Oeste dos EUA)	us-gov-west-1	peças. us-gov-west-1.amazonaws.com dicas de peças. us-gov-west-1.amazonaws.com dicas de peças. us-gov-west-1.api.aws peças. us-gov-west-1.api.aws	HTTPS

Cotas de serviço

Nome	Padrão	Ajustável	Descrição
Clusters	5	Sim	O número máximo de clusters por Região da AWS.

> **Note**
>
> Os valores padrão são as cotas iniciais definidas por AWS. Esses valores padrão são separados do valor real da cota aplicada e das cotas de serviço máximas possíveis. Para obter mais informações, consulte Terminologia do Service Quotas no Guia do usuário do Service Quotas.

Essas cotas de serviço estão listadas em Serviço de Computação AWS Paralela (PCS) no AWS Management Console. Para solicitar um aumento de cota para valores que são mostrados como ajustáveis, consulte Solicitando um aumento de cota no Guia do usuário de Cotas de Serviço.

> **Important**
>
> Lembre-se de verificar a Região da AWS configuração atual no AWS Management Console.

Cotas internas

As cotas a seguir são internas e não ajustáveis.

Nome	Padrão	Ajustável	Descrição
Criação simultânea de clusters	1	Não	O número máximo de clusters no `Creating` estado por Região da AWS.
Grupos de nós de computação por cluster	10	Não	O número máximo de grupos de nós de computação por cluster.
Filas por cluster	10	Não	O número máximo de filas por cluster.

Cotas relevantes para outros serviços AWS

AWS O PCS usa outros AWS serviços. Suas cotas de serviço para esses serviços afetam seu uso do AWS PCS.

Cotas EC2 de serviços da Amazon que afetam AWS o PCS

- Solicitações de instância spot
- Executando instâncias sob demanda
- Modelos de inicialização
- Versões do modelo de execução
- Solicitações EC2 de API da Amazon

Para obter mais informações, consulte as cotas EC2 de serviços da Amazon no Guia do usuário do Amazon Elastic Compute Cloud.

Solução de problemas no serviço de computação AWS paralela

Os tópicos a seguir fornecem orientação para solucionar alguns problemas que você pode encontrar no AWS PCS.

Tópicos

Uma EC2 instância no AWS PCS é encerrada e substituída após a reinicialização

Visão geral do problema

Depois que uma EC2 instância em um grupo de nós de computação é reinicializada, o AWS PCS encerra e substitui automaticamente a instância.

Por que isso acontece

AWS O PCS não suporta reinicializações de instâncias. Se uma EC2 instância for reinicializada, o AWS PCS considerará a instância não íntegra e a substituirá. Se o AWS PCS encerra e substitui continuamente suas instâncias, pode ser porque algo reinicializa suas instâncias após a inicialização. Alguns exemplos incluem reinicializações por automação na EC2 instância (como reinicialização automática após aplicação de patches), automação externa à EC2 instância (como um aplicativo de gerenciamento de rede), outro AWS serviço (como AWS Systems Manager) ou reinicialização manual por uma pessoa.

O que fazer

Você pode verificar seus `slurmd` registros `slurmctld` ou para ver se sua instância foi reinicializada. Para ter mais informações, consulte Logs do agendador no AWS PCS e Monitoramento de instâncias AWS PCS usando a Amazon CloudWatch. O exemplo de entrada de `slurmctld` registro a seguir indica que a instância foi reinicializada:

Example

```
[2024-09-12T06:42:50.393+00:00] validate_node_specs: Node Login-1 unexpectedly rebooted
 boot_time=1726123354 last response=1726123285
```

Reinicializando devido à aplicação de patches

Geralmente, é necessária uma reinicialização após a aplicação dos patches. Não aplique patches diretamente a uma EC2 instância que faz parte de um grupo de nós de computação do AWS PCS. Se precisar corrigir suas EC2 instâncias, você deve aplicar seus patches a uma Amazon Machine Image (AMI) atualizada e atualizar seus grupos de nós de computação para usar a AMI atualizada. EC2 As novas instâncias que o AWS PCS executa para esses grupos de nós de computação usarão a AMI atualizada (corrigida). Para obter mais informações, consulte Imagens personalizadas da Amazon Machine (AMIs) para AWS PCS.

Histórico de documentos do Guia do usuário do AWS PCS

A tabela a seguir descreve as mudanças importantes na documentação do AWS PCS.

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
3 de julho de 2025	AWS PCS lançado na Europa (Londres)	AWS O PCS agora está disponível na Europa (Londres) (eu-west-2). CloudFormation modelos estão disponíveis para começar na Europa (Londres) Região da AWS. Para obter mais informações, consulte Use AWS CloudFormation para criar um cluster AWS PCS de amostra e AWS CloudFormation modelos para criar um cluster AWS PCS de amostra.	N/D
1 de julho de 2025	Instruções atualizadas do console	Agora você pode fazer com que o AWS PCS crie um perfil de instância básico e um grupo de segurança para você ao criar um cluster e um grupo de nós de computação no console. Para obter mais informações, consulte:	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
		• Criando um cluster no AWS Parallel Computing Service • Criação de um grupo de nós de computação no AWS PCS • Perfis de instância do IAM para o AWS Parallel Computing Service	
23 de junho de 2025	Nova política gerenciada: AWSPCSCompute NodePolicy	Foi adicionada uma nova política gerenciada que concede permissão aos nós de computação do AWS PCS para se conectarem aos clusters do AWS PCS. Para obter mais informações, consulte AWS política gerenciada: AWSPCSCompute NodePolicy.	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
19 de junho de 2025	Novo tópico: registros de conclusão do trabalho	Use registros de conclusão do trabalho para registrar detalhes sobre os trabalhos quando eles forem concluídos, sem custo adicional. Para obter mais informações, consulte Registros de conclusão de trabalhos no AWS PCS.	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
18 de junho de 2025	AWS Lançamento do PCS em AWS GovCloud (US)	AWS O PCS agora está disponível em AWS GovCloud (Leste dos EUA) (us-gov-east-1) e AWS GovCloud (Oeste dos EUA) (us-gov-west-1). CloudFormation modelos estão disponíveis para começar no AWS GovCloud (US) Regions. Para obter mais informações, consulte Use AWS CloudFormation para criar um cluster AWS PCS de amostra e AWS CloudFormation modelos para criar um cluster AWS PCS de amostra. Para obter mais informações sobre os endpoints do serviço AWS PCS em AWS GovCloud (US) Regions, consulteEndpoints e cotas de serviço para PCS AWS. Para obter mais informações sobre as diferenças em AWS	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
		GovCloud (US) Regions, consulte AWS PCS AWS GovCloud (US) no Guia AWS GovCloud (US) do Usuário.	
18 de junho de 2025	Agente PCS atualizado	Atualizado o tópico da AMI para o agente AWS PCS 1.2.1-1. Para obter mais informações, consulte Instaladores de software para criar de forma personalizada AMIs para AWS PCS. >>>>>>> público	N/D
15 de maio de 2025	Novo recurso: contabilidade	A contabilidade do Slurm agora é compatível com o Slurm 24.11 ou posterior. Para obter mais informações, consulte Contabilidade de slurm no PCS AWS.	SDK DA AWS: 15/05/2015

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
15 de maio de 2025	Atualizado para o Slurm 24.11	Atualizou o guia do usuário para suporte ao Slurm 24.11.5. Para obter mais informações, consulte: • Versões Slurm no PCS AWS • Instaladores de software para criar de forma personalizada AMIs para AWS PCS • Notas de lançamento da amostra AWS PCS AMIs	N/D
5 de maio de 2025	Perguntas frequentes sobre as versões atualizadas do Slurm	Perguntas frequentes (FAQ) das versões do Slurm atualizadas sobre versões do Slurm próximas ou além do fim da vida útil (EOL). Para obter mais informações, consulte Perguntas frequentes sobre as versões do Slurm no PCS AWS.	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
17 de abril de 2025	Novo tópico: como obter detalhes do grupo de nós de computação	Saiba como obter detalhes de um grupo de nós de computação do AWS PCS, como ID, ARN e ID de AMI. Para obter mais informações, consulte Obtenha detalhes do grupo de nós de computação no AWS PCS.	N/D
2 de abril de 2025	Instalador Slurm atualizado	Atualizado o tópico da AMI para o instalador do Slurm 24.05.7-1. Para obter mais informações, consulte Instaladores de software para criar de forma personalizada AMIs para AWS PCS.	N/D
28 de março de 2025	Foram adicionadas cotas para o número máximo de grupos e filas de nós de computação	Foram adicionadas cotas internas não ajustáveis para o número máximo de grupos de nós de computação por cluster e o número máximo de filas por cluster. Para obter mais informações, consulte Cotas internas.	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
14 de março de 2025	Alterou uma chave de propriedade no CloudFormation modelo	`Id` agora é `TemplateId` para a `CustomLaunchTemplate` propriedade no CloudFormation modelo. Para obter mais informações, consulte Recursos em Partes de um CloudFormation modelo para AWS PCS.	N/D
13 de março de 2025	Informações de versão adicionadas para o agente AWS PCS e o Slurm	Foi adicionado um novo tópico que descreve as alterações em cada versão do agente AWS PCS. Para obter mais informações, consulte AWS Versões do agente PCS. Foram adicionadas mais informações ao tópico de versões do Slurm que descreve datas de suporte importantes e notas de lançamento detalhadas para o suporte do AWS PCS para o Slurm. Para obter mais informações, consulte Versões Slurm no PCS AWS.	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
07 de março de 2025	Agente PCS atualizado	Atualizado o tópico da AMI para o agente AWS PCS 1.2.0-1. Para obter mais informações, consulte Instaladores de software para criar de forma personalizada AMIs para AWS PCS.	N/D
3 de fevereiro de 2025	Foi adicionado um tópico sobre o uso AWS CloudFormation com o AWS PCS	Foi adicionado um tópico ao guia do usuário que fornece um exemplo de como usar AWS CloudFormation com o AWS PCS. O tópico fornece um procedimento para usar um CloudFormation modelo de amostra para criar o cluster AWS PCS de amostra e descreve resumidamente as seções desse modelo. Para obter mais informações, consulte Comece a usar um AWS CloudFormationAWS PCS.	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
18 de dezembro de 2024	Atualizado para o Slurm 24.05	Atualizou o guia do usuário para suporte ao Slurm 24.05. Para obter mais informações, consulte Instaladores de software para criar de forma personalizada AMIs para AWS PCS e Notas de lançamento da amostra AWS PCS AMIs.	N/D
18 de dezembro de 2024	Versões atualizadas da NVIDIA para a amostra Slurm 23.11 AMIs	O driver NVIDIA e as versões CUDA foram atualizados na amostra do Slurm 23.11. AMIs Para obter mais informações, consulte Notas de lançamento da amostra AWS PCS AMIs.	N/D
17 de dezembro de 2024	Instalador Slurm atualizado	Atualizado o tópico da AMI para o instalador do Slurm 23.11.10-3. Para obter mais informações, consulte Instaladores de software para criar de forma personalizada AMIs para AWS PCS.	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
13 de dezembro de 2024	Agente PCS atualizado	Atualizado o tópico da AMI para o agente AWS PCS 1.1.1-1. Para obter mais informações, consulte Instaladores de software para criar de forma personalizada AMIs para AWS PCS.	N/D
06 de dezembro de 2024	Agente PCS atualizado e instalador do Slurm	O tópico da AMI foi atualizado para o agente AWS PCS 1.1.0-1 e o instalador do Slurm 23.11.10-2. Para obter mais informações, consulte Instaladores de software para criar de forma personalizada AMIs para AWS PCS.	N/D
06 de dezembro de 2024	Foi adicionado um tópico sobre suporte ao sistema operacional	Para obter mais informações, consulte Sistemas operacionais compatíveis no AWS PCS.	N/D
8 de novembro de 2024	Guia do usuário reorganizado	Reorganizamos o guia do usuário para colocar os tópicos no nível superior, movemos alguns tópicos para suas próprias páginas e agrupamos tópicos semelhantes.	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
7 de novembro de 2024	Tópicos atualizados da AMI	Atualizado o tópico da AMI para o Slurm 23.11.10 e libjwt 17.0. Para obter mais informações, consulte Instaladores de software para criar de forma personalizada AMIs para AWS PCS e Etapa 3 — Instalar o Slurm. Simplificou e corrigiu as notas de lançamento do. AMIs Para obter mais informações, consulte Notas de lançamento da amostra AWS PCS AMIs.	N/D
7 de novembro de 2024	Foi adicionado um novo tópico sobre o uso de volumes criptografados do EBS com AWS o PCS	Foi adicionado um tópico que descreve a política de chaves do KMS necessária para volumes criptografados do EBS no AWS PCS. Para obter mais informações, consulte Política de chave KMS necessária para uso com volumes criptografados do EBS no PCS AWS.	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
18 de outubro de 2024	AWS Lançado o agente PCS 1.0.1-1	Documentação relacionada à AMI atualizada para se referir à versão 1.0.1-1 do agente AWS PCS. Para obter mais informações, consulte Instaladores de software para criar de forma personalizada AMIs para AWS PCS e Etapa 2 — Instalar o agente AWS PCS.	N/D
10 de outubro de 2024	Foi adicionado um capítulo de solução de problemas	Foi adicionado um capítulo de solução de problemas com um tópico sobre a substituição automática de EC2 instâncias após a reinicialização. Para obter mais informações, consulte Solução de problemas no serviço de computação AWS paralela.	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
23 de setembro de 2024	Atualizou as permissões mínimas para usar ações de API e para um administrador de serviços	Agora, a `ec2:DescribeInstanceTypeOfferings` permissão é necessária para `CreateComputeNodeGroup` as ações `UpdateComputeNodeGroup` da API. Para obter mais informações, consulte Permissões mínimas para AWS PCS.	N/D
5 de setembro de 2024	Atualizou o exemplo de política do IAM para as permissões mínimas para um administrador de serviços	Para obter mais informações, consulte Permissões mínimas para um administrador de serviços.	N/D
5 de setembro de 2024	Foi adicionada uma permissão ausente ao JSON na página de políticas gerenciadas	Essa foi apenas uma correção na documentação. A política gerenciada real não foi alterada. Para obter mais informações, consulte AWS políticas gerenciadas para o Serviço de Computação AWS Paralela.	N/D

Data	Alteração	Atualizações feitas na documentação	Versões de API atualizadas
28 de agosto de 2024	Página de políticas gerenciadas adicionada	Para obter mais informações, consulte AWS políticas gerenciadas para o Serviço de Computação AWS Paralela.	N/D
28 de agosto de 2024	AWS Lançamento do PCS	Versão inicial do guia do usuário do AWS PCS.	AWS SDK: 2024-08-28

AWS Glossário

Para obter a AWS terminologia mais recente, consulte o AWS glossário na Glossário da AWS Referência.

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.