Manual do usuário

AWS PCS

AWS PCS: Manual do usuário

Table of Contents

O que é serviço de computação AWS paralela?

AWS O Parallel Computing Service (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 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.

Principais conceitos

Um cluster em 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 HPC arquiteturas sofisticadas.

Cluster

Um cluster é um recurso para gerenciar recursos e executar cargas de trabalho. Um cluster é um AWS PCS recurso 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 deseja que os recursos do cluster sejam lançados. O agendador aceita e agenda trabalhos e também inicia os nós de computação (EC2instâncias) que processam esses trabalhos.

Grupo de nós de computação

Um grupo de nós de computação é uma coleção de nós de computação AWS PCS usados 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, VPC sub-redes de destino, Amazon Machine Image (AMI), opção de compra e configuração de lançamento personalizada. AWS PCSusa 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 seja AWS PCS programado 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 AWS PCS por meio do AWS Management Console AWS PCSAPI, AWS SDK e. 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 (comoSSH) para acessar recursos do cluster, executar trabalhos, gerenciar dados e realizar outras atividades baseadas em shell.

Configurando o serviço de computação AWS paralela

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 e inserir um código de verificação no 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 atributos 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 sua 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. Ative a autenticação multifator (MFA) para seu usuário root.

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

Criar um usuário com acesso administrativo

1. Ative o IAM Identity Center.

   Para obter instruções, consulte Habilitar AWS IAM Identity Center no Guia do usuário do AWS IAM Identity Center .
2. No IAM Identity Center, conceda 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 entrar com seu usuário do IAM Identity Center, use o login URL que foi enviado ao seu endereço de e-mail quando você criou o usuário do IAM Identity Center.

  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 IAM Identity Center, crie um conjunto de permissões que siga as melhores práticas de aplicação de permissões com privilégios mínimos.

Para obter instruções, consulte Create a permission set 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 Add groups no Guia do usuário do AWS IAM Identity Center .

Instale o AWS CLI

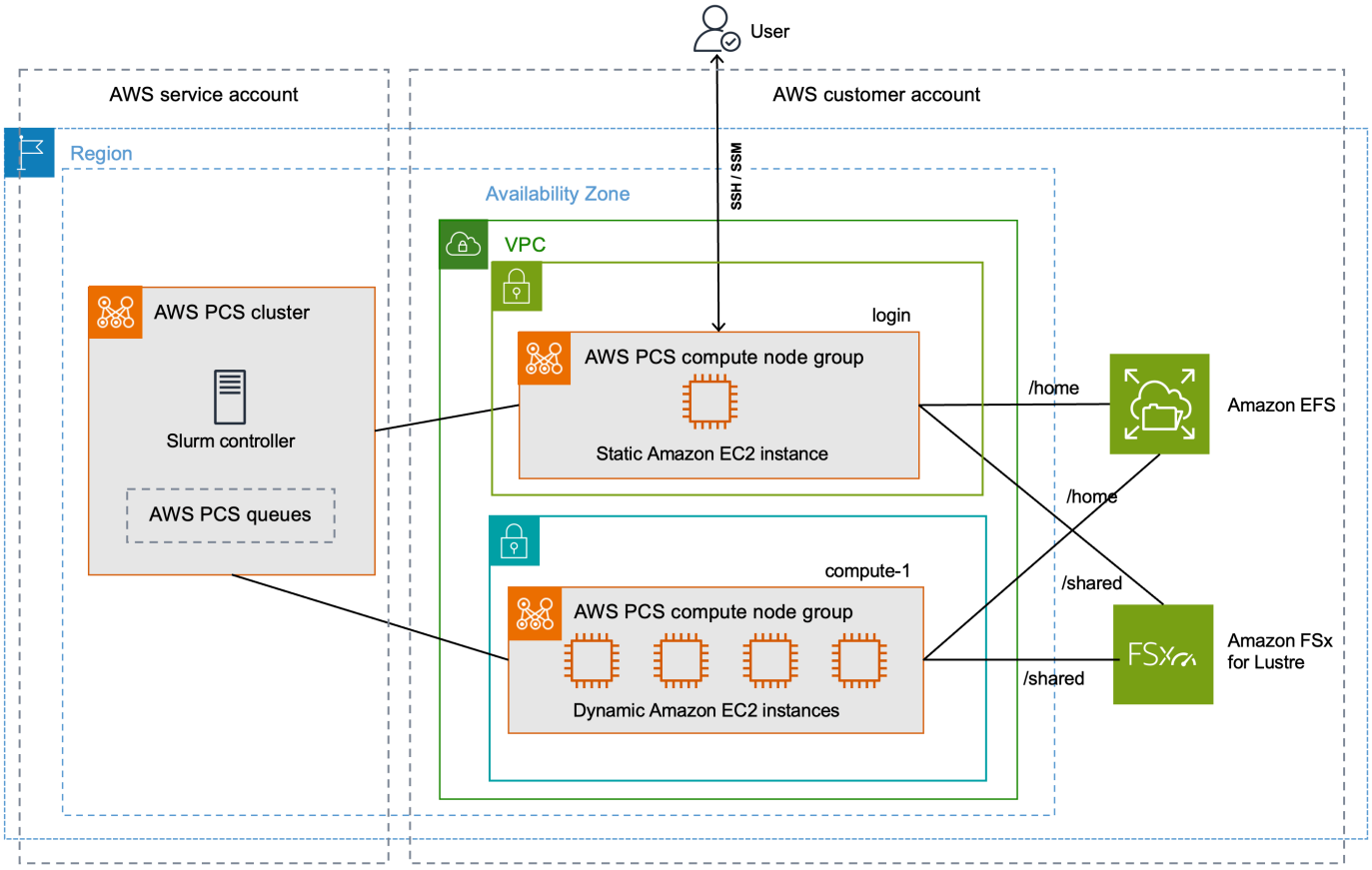
Você deve usar a versão mais recente do AWS CLI. Para obter informações, consulte Instalar ou atualizar para a versão mais recente do AWS CLI no Guia AWS Command Line Interface do Usuário da Versão 2.

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

```
aws pcs help
```

Começando com AWS PCS

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



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

- A VPC e sub-redes que atendem aos requisitos AWS PCSde rede.
- Um sistema de EFS arquivos da Amazon, 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 AWS PCS cluster, que fornece um controlador Slurm.
- 2 grupos de nós de computação.
  - 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 requer AWS recursos adicionais, como grupos de segurança, IAM funções e modelos de EC2 execução, que não são mostrados no diagrama.

Tópicos

Pré-requisitos para começar a usar AWS PCS

Antes de começar este tutorial, instale e configure as seguintes ferramentas e recursos necessários para criar e gerenciar um AWS PCS cluster.

- AWS CLI— Uma ferramenta de linha de comando para trabalhar com AWS serviços, inclusive AWS PCS. 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. Depois de instalar o AWS CLI, recomendamos que você também o configure. Para obter mais informações, consulte Configurar o AWS CLI no Guia AWS Command Line Interface do usuário para a versão 2.
- IAMPermissões necessárias — O diretor de IAM segurança que você está usando deve ter permissões para trabalhar com AWS PCS IAM funções, funções vinculadas ao serviço AWS CloudFormation,VPC, a 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
```

- 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.

Crie uma VPC e sub-redes para AWS PCS

Você pode criar sub-redes a VPC e com um CloudFormation modelo. Use o seguinte URL 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:
  - Abaixo de 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 IAM recursos”.

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

Encontre o grupo de segurança padrão para o cluster VPC

Para encontrar a ID do grupo de segurança padrão no novoVPC, siga este procedimento:

- Navegue até o VPCconsole da Amazon.

- No VPCPainel, selecione Filtrar por VPC.
  - Escolha o VPC local com o 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 PCSdepende de grupos de segurança para gerenciar o tráfego de rede que entra e sai de um cluster e de 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 dois grupos de segurança.

- Um grupo de segurança de cluster, que permite a comunicação entre o AWS PCS controlador, os nós de computação e os nós de login.
- Um grupo de SSH segurança 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 criar sub-redes a VPC e com esse CloudFormation modelo. Use o seguinte URL 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 o VPC local com o qual o nome começa`hpc-networking`.
  - (Opcional) Em ClientIpCidr, insira um intervalo de IP mais restritivo para o grupo de SSH segurança 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. PUBLIC 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 de entrada SSH

Crie um cluster em AWS PCS

Em AWS PCS, um cluster é um recurso persistente para gerenciar recursos e executar cargas de trabalho. Você cria um cluster para um agendador específico (AWS PCSatualmente suporta Slurm) em uma sub-rede nova ou existente. VPC O cluster aceita e agenda trabalhos e também inicia os nós de computação (EC2instâncias) que processam esses trabalhos.

Para criar um cluster

1. Abra o AWS PCSconsole e escolha Criar cluster.
2. Na seção Configuração do cluster, insira os seguintes campos:
   - Nome do cluster — Enter `get-started`
   - Tamanho do controlador — Selecione Pequeno
3. Na seção Rede, selecione valores para os seguintes campos:
   - VPC— Escolha o VPC nome `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 o AWS PCS Amazon Elastic File System

O Amazon Elastic File System (AmazonEFS) 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 AWS PCS demonstração usa um sistema de EFS arquivos para fornecer um diretório inicial compartilhado entre os nós do cluster. Crie um sistema de EFS arquivos VPC igual ao seu cluster.

Para criar seu sistema de EFS arquivos da Amazon

1. Acesse o EFSconsole da Amazon.
2. Certifique-se de que esteja configurado da mesma forma Região da AWS em que você tentará 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 o VPC nome `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 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. Você usa o Lustre para cargas de trabalho em que a velocidade é importante, como aprendizado de máquina, computação de alto desempenho (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 AWS PCS demonstração pode usar um sistema de arquivos FSx for Lustre para fornecer um diretório compartilhado de alto desempenho entre os nós do cluster. Crie um sistema de arquivos FSx for Lustre no mesmo VPC que seu cluster.

Para criar seu sistema de arquivos FSx for Lustre

1. Acesse o FSxconsole 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 o VPC nome `hpc-networking:Large-Scale-HPC`
    - Para Grupos VPC de Segurança, 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 em AWS PCS

Um grupo de nós de computação é uma coleção virtual de nós de computação (EC2instâncias) que são executados e AWS PCS gerenciados. Ao definir um grupo de nós de computação, você especifica características comuns, como tipos de EC2 instância, contagem mínima e máxima de instâncias, VPC sub-redes de destino, opção de compra preferencial e configuração de execução personalizada. AWS PCSinicia, 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
- Crie modelos de lançamento para AWS PCS

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

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 AmazonEC2, 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 AmazonEC2, 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 função.
- 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 AWSPCSde -getstarted-policy para adicioná-la à função.
  - Em Políticas de permissões, pesquise por mazonSSMManaged InstanceCore A.
  - Marque a caixa ao lado mazonSSMManagedInstanceCorede A para adicioná-la à função.
  - Escolha Próximo.
- Em Nome, revise e crie:
  - Em Detalhes da função:
    - Em Nome do perfil, insira `AWSPCS-getstarted-role`.
  - Escolha Create role (Criar função).

Crie modelos de lançamento para AWS PCS

Ao criar um grupo de nós de computação, você fornece um modelo de EC2 execução que 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 SSH acesso de entrada.

Acesse o CloudFormation modelo

Use o seguinte URL 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 clusterVPC.
    - 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 SSH chaves preferido.
  - Em Sistemas de arquivos
    - Para EfsFilesystemId, insira a ID do sistema de EFS arquivos do sistema de arquivos 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 (EC2instâncias) que são executados e AWS PCS gerenciados. Ao definir um grupo de nós de computação, você especifica características comuns, como tipos de EC2 instância, contagem mínima e máxima de instâncias, VPC sub-redes de destino, opção de compra preferencial e configuração de execução personalizada. AWS PCSinicia, 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 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 AWS PCSconsole 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:
  - EC2modelo de lançamento — Escolha o modelo de lançamento em que o nome está `login-getstarted-lt`
  - IAMperfil da instância — 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:
  - AMIID — Selecione o AMI local com o qual o nome começa `aws-pcs-sample_ami-amzn2-x86_64-slurm-23.11`
- 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 AWS PCS

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 AWS PCSconsole 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:
  - EC2modelo de lançamento — Escolha o modelo de lançamento em que o nome está `compute-getstarted-lt`
  - IAMperfil da instância — 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:
  - AMIID — Selecione o AMI local com o qual o nome começa`aws-pcs-sample_ami-amzn2-x86_64-slurm-23.11`.
- 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 seja AWS PCS programado 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 AWS PCSconsole.
- 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 AWS PCS cluster

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 AWS PCSconsole 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 EC2console da Amazon.
  - Selecione Instances (Instâncias).
  - Pesquise EC2 instâncias com a seguinte tag. Substituir *`node-group-id`* com o 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 ouSSH.

    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 Conectar à instância, selecione SSHcliente.
- Siga as instruções fornecidas pelo console.

> **Note**
>
> O nome de usuário da instância **ec2-user**não éroot.

Explore o ambiente de cluster em 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 comossm-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 usandoSSH.

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

Trabalhe com sistemas de arquivos compartilhados

Você pode confirmar se o sistema de EFS arquivos 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 EFS arquivos 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 uso`sinfo`. 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 chamada`demo`. Seu status é `up` e tem no máximo 4 nós. Ele 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 lida`compute-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 com`squeue`. 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 um trabalho 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 no prompt do 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. Substituir *`job-id`* no comando a seguir com o número retornado de`sbatch`.

```
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 um MPI trabalho de vários nós com o Slurm in AWS PCS

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

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 MPI módulo Open.

```
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 AWS PCS recursos que você criou para este tutorial

- Abra o AWS PCSconsole.
- 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 IAMconsole.
  - 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 EFSconsole da Amazon.
  - 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 FSxconsole 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.

Trabalhando com AWS PCS

Este capítulo fornece informações e orientações para ajudá-lo a usar AWS PCS.

Tópicos

AWS PCSaglomerados

Um AWS PCS cluster consiste nos seguintes componentes:

- Instâncias gerenciadas do software programador HPC do sistema, como o daemon de controle Slurm (). `slurmctld`
- Componentes que se integram ao agendador HPC do sistema para provisionar e gerenciar EC2 instâncias da Amazon.
- Componentes que se integram ao programador HPC do sistema 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 PCSprovisiona interfaces de rede elásticas em sua VPC sub-rede da Amazon 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 AWS PCS cluster, recomendamos que você sigaComeçando com AWS PCS. O tutorial pode ajudá-lo a criar um HPC sistema funcional sem expandir para todas as opções disponíveis e arquiteturas de sistema possíveis.

Pré-requisitos

- Uma sub-rede existente VPC e que atende aos AWS PCSTrabalho em rede requisitos. Antes de implantar um cluster para uso em produção, recomendamos que você tenha uma compreensão completa dos requisitos VPC e da sub-rede. Para criar uma sub-rede VPC e, consulteCriando um VPC para seu AWS PCS cluster.
- Um IAMdiretor com permissões para criar e gerenciar AWS PCS recursos. Para obter mais informações, consulte Identity and Access Management for AWS Parallel Computing Service.

Crie um AWS PCS cluster

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

AWS Management Console

Para criar um cluster

1. Abra o AWS PCS console 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. AWS PCSatualmente suporta o Slurm 23.11. Para obter mais informações, consulte Versões do Slurm em AWS PCS.
- Tamanho do controle — Escolha um tamanho para o controle. Isso determina quantos trabalhos e nós de computação simultâneos podem ser gerenciados pelo AWS PCS cluster. Você só pode definir o tamanho do controlador quando o cluster é criado. Para obter mais informações sobre dimensionamento, consulteEscolhendo um tamanho AWS PCS de cluster.

3. Na seção Rede, selecione valores para os seguintes campos:

   - VPC— Escolha um existente VPC que atenda aos AWS PCS requisitos. Para obter mais informações, consulte AWS PCSVPCe requisitos e considerações de sub-rede. Depois de criar o cluster, você não pode alterá-loVPC. Se nenhum VPCs estiver listado, você deverá criar um primeiro.
   - Sub-rede — Todas as sub-redes disponíveis na selecionada VPC são listadas. Escolha dois em diferentes zonas de disponibilidade. Cada sub-rede deve atender aos requisitos da AWS PCS sub-rede. Para obter mais informações, consulte AWS PCSVPCe requisitos e considerações de 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 associar AWS PCS 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. Para obter mais informações, consulte Requisitos e considerações do grupo de segurança.

4. (Opcional) Em Criptografia, você pode definir uma chave personalizada para criptografar os dados do controlador definindo estes campos:

   - KMSID da chave — Deixe como `aws/pcs` usar a KMS chave que PCS cria. Selecione um alias de KMS chave existente para usar uma KMS chave personalizada. Observe que a conta usada para criar o cluster deve ter `kms:Decrypt` privilégios na KMS chave personalizada.

5. (Opcional) Na seção Configuração do Slurm, você pode especificar as opções de configuração do Slurm que substituem os padrões definidos por: AWS PCS

- 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 HPC sistema 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` CPU ativará o agendamento somente. Esse parâmetro corresponde à SelectTypeParametersconfiguração no Slurm, onde `SelectType` é definido como por. `select/cons_tres` AWS PCS

6. (Opcional) Em Tags, adicione qualquer tag ao seu AWS PCS cluster.
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 PCSretornará 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:

   - Substituir *region* com o ID do Região da AWS qual você deseja criar seu cluster, como `us-east-1`.
   - Substituir *my-cluster* com 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 dentro Região da AWS e Conta da AWS onde você está criando o cluster.
   - Substituir *23.11* com qualquer versão compatível do Slurm.

     > **Note**
     >
     > AWS PCSatualmente suporta o Slurm 23.11.

   - Substituir *SMALL* com qualquer tamanho de cluster compatível. Isso determina quantos trabalhos e nós de computação simultâneos podem ser gerenciados pelo AWS PCS cluster. Ele só pode ser definido quando o cluster é criado. Para obter mais informações sobre dimensionamento, consulte Escolhendo um tamanho AWS PCS de cluster.
   - 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 associar AWS PCS às interfaces de rede que ele cria para o seu cluster. Os grupos de segurança devem estar no VPC mesmo 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 ajustar o comportamento do Slurm adicionando uma opção. `--slurm-configration` Por exemplo, você pode definir o tempo de inatividade de redução para 60 minutos (3600 segundos) com. `--slurm configuration scaleDownIdeTime=3600`
   - Opcionalmente, você pode fornecer uma KMS chave personalizada para criptografar os dados do seu controlador usando. `--kms-key-id` *kms-key* *kms-key*Substitua por um

ID de chave ou alias existente KMSARN. Observe que a conta usada para criar o cluster deve ter `kms:Decrypt` privilégios na KMS chave personalizada.

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

2. 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 PCSretornará 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 AWS PCS cluster.

Considerações ao excluir um cluster AWS 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 em AWS PCS.

- 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 AWS PCS.

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 AWS PCSconsole.
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:
   - Substituir *region-code* com o em que Região da AWS seu cluster está.
   - Substituir *my-cluster* com o 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
```

Escolhendo um tamanho AWS PCS de cluster

AWS PCSfornece 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 filas que você espera executar no 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 em AWS PCS

Como parte da criação de um cluster, 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 AWS PCS computação, que definem conjuntos de instâncias a serem executadas em resposta a eventos de escalabilidade. AWS PCSconfigura 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 PCSarmazena o segredo do cluster como um segredo gerenciado com o prefixo `pcs!` in AWS Secrets Manager. O custo do segredo está incluído na taxa de uso AWS PCS.

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

Sumário

Encontre o segredo do cluster Slurm

Você pode encontrar segredos AWS PCS gerenciados usando o AWS Secrets Manager console ouAPI, diretamente de AWS PCS, ou usando tags.

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 AWS PCS cluster tem um nome no formato em `pcs!slurm-secret-`*`cluster-id`* que *`cluster-id`* está o ID do AWS PCS cluster.

AWS CLI

Cada segredo AWS PCS do cluster 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, com`us-east-1`.
- *`cluster-id`*Substitua pelo ID do AWS PCS cluster 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 AWS PCS para encontrar o segredo do cluster

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

- *`region`*Substitua pelo Região da AWS para criar seu cluster, com`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": "pcsdemo",
        "id": "s3431v9rx2",
        "arn": "arn:aws:pcs:us-east-1:012345678901:cluster/s3431v9rx2",
        "status": "ACTIVE",
        "createdAt": "2024-07-12T15:32:27.225136+00:00",
        "modifiedAt": "2024-07-12T15:32:27.225136+00:00",
        "scheduler": {
            "type": "SLURM",
            "version": "23.11"
```

```
        },
        "size": "SMALL",
        "networking": {
            "subnetIds": [
                "subnet-0123456789abcdef"
            ],
            "securityGroupIds": [
                "sg-0123456789abcde"
            ]
        },
        "endpoints": [
            {
                "type": "SLURMCTLD",
                "privateIpAddress": "127.0.0.1",
                "port": "6817"
            }
        ],
        "secretArn": "arn:aws:secretsmanager:us-east-1:012345678901:secret:pcs!slurm-
secret-s3431v9rx2-FN7tJF",
        "secretVersion": "ff58d1fd-070e-4bbc-98a0-64ef967cebcc"
    }
}
```

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 AWS PCS cluster.

```
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 AWS PCS login

Permissões

Você usa um IAM principal para obter o segredo do cluster Slurm. O IAM diretor 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 IAM política a seguir permite 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 PCSgrupos de nós de computação

Um grupo de nós de AWS PCS computação é uma coleção lógica de nós (EC2instâncias da Amazon). Esses nós podem ser usados para executar trabalhos de computação, bem como para fornecer acesso interativo baseado em shell a um HPC sistema. 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, 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
- Atualização de um grupo de nós AWS PCS de computação
- Excluindo um grupo de nós de computação no AWS PCS

- Encontrando instâncias de grupos de nós de computação em AWS PCS

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 AWS Parallel Computing Service (AWS PCS). Se esta é a primeira vez que você cria um grupo de nós de computação em AWS PCS, recomendamos que você siga o tutorial emComeçando com AWS PCS. O tutorial pode ajudá-lo a criar um HPC sistema 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 sub-rede existente VPC e uma (s) que atendam aos requisitos AWS PCS de rede. Recomendamos que você entenda completamente esses requisitos antes de implantar um cluster para uso em produção. Para obter mais informações, consulte AWS PCSVPCe requisitos e considerações de sub-rede. Você também pode usar um CloudFormation modelo para criar sub-redes VPC e. AWS fornece uma HPC receita para o CloudFormation modelo. Para obter mais informações, consulte aws-hpc-recipesem GitHub.
- Um perfil de IAM instância com permissões para chamar a AWS PCS `RegisterComputeNodeGroupInstance` API ação e acessar quaisquer outros AWS recursos necessários para as instâncias do seu grupo de nós. Para obter mais informações, consulte IAMperfis de instância 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 AWSServiceRoleForEC2Spotvinculada ao serviço 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 AWS PCSconsole.
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. EC2modelo 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 PCScria 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. EC2versão do modelo de lançamento — Selecione uma versão do seu modelo de lançamento personalizado. Você pode escolher uma versão específica, que pode melhorar a reprodutibilidade. 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 AWS PCS de computação.

   c. AMIID — se seu modelo de lançamento não incluir um AMI ID ou se você quiser substituir o valor no modelo de lançamento, forneça um AMI ID aqui. Observe que o AMI usado para o grupo de nós deve ser compatível com AWS PCS. Você também pode

selecionar uma amostra AMI fornecida por AWS. Para obter mais informações sobre esse tópico, consulteAmazon Machine Images (AMIs) para AWS PCS.

d. IAMperfil de instância — 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, veja IAMperfis de instância para o AWS Parallel Computing Service para aprender como criar um.

e. Sub-redes — Escolha uma ou mais sub-redes no local em VPC que seu AWS PCS cluster está implantado. Se você selecionar várias sub-redes, as EFA comunicações 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 (x864_64 ou arm64) e número de. vCPUs Se as instâncias tiveremGPUs, todos os tipos de instância deverão ter o mesmo número deGPUs.

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 PCSNão 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 do cluster em AWS PCS. Para obter mais informações, consulte RealMemorya Slurm documentação.

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 AWS PCS provisiona o grupo de nós. Isso pode demorar vários minutos.

Próxima etapa recomendada

- Adicione seu grupo de nós a uma fila 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. Substituir *`region`*com o ID do Região da AWS para criar seu cluster, com`us-east-1`.
2. Substituir *`my-cluster`* com o nome ou `clusterId` do seu cluster.
3. Substituir *`my-node-group`*com o 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. Substituir *`subnet-ExampleID1`* com uma ou mais sub-redes IDs do seu cluster. VPC
5. Substituir *`lt-ExampleID1`* com o 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 PCScria 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. Substituir *launch-template-version* com uma versão específica do modelo de lançamento se você quiser associar seu grupo de nós a uma versão específica.
7. Substituir *arn:InstanceProfile*com o perfil ARN da sua IAM instância. Se você não tiver um preparado, consulte Usando modelos de EC2 lançamento da Amazon com AWS PCS para obter orientação.
8. Substituir *min-instances* e *max-instances* com 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 PCSNão oferece suporte a grupos de nós de computação com uma combinação de instâncias estáticas e dinâmicas.
9. Substituir *t3.large* com 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 (x864_64 ou arm64) e número de. vCPUs Se as instâncias tiveremGPUs, todos os tipos de instância deverão ter o mesmo número deGPUs.

```
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 um AMI ou se você deseja substituir esse valor. Observe que o AMI usado para o grupo de nós deve ser compatível com AWS PCS. Você também pode selecionar uma amostra

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, você também pode usar `--allocation-strategy` para definir como AWS PCS escolhe os pools de capacidade spot quando executa 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 AWS PCS em sua Slurm configuração. Para obter mais informações, consulte RealMemorya Slurm documentação.

> **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 AWS PCS de computação

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

Opções para atualizar um grupo de nós AWS PCS de computação

A atualização de um grupo de nós de AWS PCS computação permite que você altere as propriedades das instâncias lançadas por AWSPCS, bem como as regras de como essas instâncias

são executadas. Por exemplo, você pode substituir as instâncias do grupo de quatro nós AMI 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:

- Nome
- Instâncias

Considerações ao atualizar um grupo de nós de AWS PCS 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 AWS PCS filas. 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 IAM instância usado por um grupo de nós de computação, isso terá efeito imediato nas instâncias em execução e no futuro.
- Para alterar o AMI uso das instâncias de um grupo de nós de computação, atualize o grupo de nós de computação (ou seu modelo de execução) para usar o novo AMI e aguarde AWS PCS a substituição das instâncias.
- AWS PCSsubstitui 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 de AWS PCS substituir 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 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, AWS PCS removerá os nós do Slurm para atingir o novo máximo. AWS PCSencerra as instâncias em execução associadas aos nós do Slurm removidos. Os trabalhos em execução nos nós removidos falham e retornam às filas.
- AWS PCScria 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 AWSServiceRoleForEC2Spotvinculada ao serviço em sua conta. Para obter mais informações, consulte Função do Amazon EC2 Spot para AWS PCS.

Para atualizar um grupo de nós AWS PCS de computação

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

AWS Management Console

Para atualizar um grupo de nós de computação

1. Abra o AWS PCS console 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. Nas seções Configuração de computação, Configurações adicionais e Configurações de Slurmpersonalização, 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. Substituir *region-code* com a AWS região na qual você deseja criar seu cluster.
   b. Substituir *my-node-group* com o nome ou `computeNodeGroupId` para seu grupo de nós de computação.
   c. Substituir *my-cluster* com o nome ou `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 AMI ID, passe `--amiId my-custom-ami-id` onde *my-custom-ami-id* é substituído pelo 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 AWS PCS

Este tópico fornece uma visão geral das opções disponíveis e descreve o que considerar ao excluir um grupo de nós de computação em 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 AWS PCS filas. 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 AWS PCS fila.

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 AWS PCSconsole.
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 o 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:
  - Substituir *region-code* com o em que Região da AWS seu cluster está.
  - Substituir *my-node-group* com o nome ou ID do seu grupo de nós de computação.
  - Substituir *my-cluster* com o 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 o Slurm.

Encontrando instâncias de grupos de nós de computação em AWS PCS

Cada grupo de nós de AWS PCS computação 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 AWS PCSconsole.
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 EC2console e escolha Instâncias.
6. Pesquise as instâncias com a seguinte tag. Substituir *node-group-id* com o ID (não o 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. Exemplo: `us-east-1`
- *node-group-id*Substitua pelo ID (não pelo nome) do seu grupo de nós de computação.
- `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 DescribeInstanceso Amazon Elastic Compute Cloud API Reference.

Usando modelos de EC2 lançamento da Amazon com AWS PCS

Na AmazonEC2, um modelo de lançamento pode armazenar um conjunto de preferências para que você não precise especificá-las individualmente ao iniciar instâncias. AWS PCSincorpora 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 PCScria 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 AWS PCS ele. 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

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 AWS PCS, mas há algumas exceções.

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

- Tipo de instância/Especificar atributos do tipo de instância (`InstanceRequirements`) — 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 de IAM instância (`IamInstanceProfile`) — Você fornece isso ao criar ou atualizar o grupo de nós.
- Detalhes avançados/Desativar a API terminação (`DisableApiTermination`) — AWS PCS deve controlar o ciclo de vida das instâncias do grupo de nós que ela executa.
- Detalhes avançados/Desabilitar API stop (`DisableApiStop`) — AWS PCS deve controlar o ciclo de vida das instâncias do grupo de nós que ele executa.
- Detalhes avançados/Stop — Hibernate behavior (`HibernationOptions`) — não AWS PCS suporta a hibernação de instâncias.
- Detalhes avançados/Elastic GPU (`ElasticGpuSpecifications`) — O 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.
- AAdvancedDetalhes/especificar CPU opções/segmentos por núcleo (`ThreadsPerCore`) — AWS PCS define o número de segmentos por núcleo como 1.

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

- Dados do usuário (`UserData`) — Isso deve ser codificado em várias partes. Consulte Trabalhando com dados de EC2 usuários da Amazon.
- Imagens do aplicativo e do sistema operacional (`ImageId`) — Você pode incluir isso. No entanto, se você especificar uma AMI ID ao criar ou atualizar o grupo de nós, ela substituirá o valor no modelo de execução. O AMI que você fornece deve ser compatível com 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 AWS PCS modelo de lançamento. 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 em AWS PCS.

- Configurações de rede/Configuração de rede avançada (`NetworkInterfaces`) — Se você usa EC2 instâncias com uma única placa de rede e não precisa de nenhuma configuração de rede especializada, 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 em 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 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 em 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 SSH chaves que será usado para fazer login nas 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 AWS PCS clusterVPC. No mínimo, escolha:
   - Um grupo de segurança que permite a comunicação com o AWS PCS cluster
   - Um grupo de segurança que permite a comunicação entre EC2 instâncias iniciadas pelo AWS PCS
   - (Opcional) Um grupo de segurança que permite SSH acesso 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 AWS PCS computação.

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. Substituir *`region-code`* com o Região da AWS local com o qual você está trabalhando AWS PCS
  b. Substituir *`my-launch-template-name`* com um nome para seu modelo. Ele deve ser exclusivo do Conta da AWS e Região da AWS que você está usando.
  c. Substituir *`my-ssh-key-name`* com o nome da sua SSH chave preferida.
  d. Substituir *`sg-ExampleID1`* e *`sg-ExampleID2`* com 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 segurançaIDs. Todos os grupos de segurança incluídos no modelo de execução devem ser do seu AWS PCS clusterVPC.

```
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 AWS PCS computação.

Trabalhando com dados de EC2 usuários da Amazon

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 no AWS PCSAPI, 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 MIME várias partes. Isso ocorre porque seus dados de usuário são mesclados com outros dados de AWS PCS usuário necessários para configurar nós em seu grupo de nós. Você pode combinar vários blocos de dados do usuário em um único arquivo MIME com várias partes.

Um arquivo MIME com várias partes consiste nos seguintes componentes:

- O tipo de conteúdo e a declaração de limite da parte: `Content-Type: multipart/mixed; boundary="==BOUNDARY=="`
- A declaração da MIME versão: `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 de MIME 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 an AWS SDK, deverá primeiro codificar em base64 os dados do usuário e enviar essa string como o valor do `UserData` parâmetro ao chamar CreateLaunchTemplate, conforme mostrado neste JSON arquivo.

```
{
    "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 EFS arquivos como um diretório inicial compartilhado

Exemplo: instalar software a AWS PCS 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.

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 a AWS PCS 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.

Esse script usa o 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 seguintes valores nesse script pelos seus próprios detalhes:

- *`my-bucket-name`* — O nome de um bucket do S3 que sua conta pode ler.
- *`path`* — O caminho relativo à raiz do bucket do S3.
- *`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://my-bucket-name/path /tmp/script.sh
- /usr/bin/shell /tmp/script.sh

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

O perfil da IAM instância do grupo de nós deve ter acesso ao bucket. A IAM política 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:::my-bucket-name",
                "arn:aws:s3:::my-bucket-name/path/*"
            ]
        }
    ]
}
```

Exemplo: definir variáveis de ambiente globais 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.

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 EFS arquivos 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.

Este exemplo estende o exemplo EFS mount in 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 EFS arquivos. 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 EFS arquivos.
- *`filesystem-id`* — O ID do sistema de EFS arquivos do sistema de arquivos.

```
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==--
```

Habilitando sem senha SSH

Você pode usar o exemplo do diretório inicial compartilhado para implementar SSH conexões entre instâncias de cluster usando SSH chaves. 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 SSH conexões entre os nós do cluster.

Reservas de capacidade em 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 PCSoferece 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 AWS PCS consome suas instâncias reservadas. Se você criar uma aberturaODCR, todas as instâncias correspondentes iniciadas por AWS PCS ou outros processos em sua conta serão contabilizadas na reserva. Com uma segmentaçãoODCR, somente as instâncias lançadas 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 AWS PCS computação para usar um alvo ODCR adicionando-o a um modelo de execução. Aqui estão as etapas para fazer isso:

1. Crie uma reserva de capacidade sob demanda direcionada (ODCR).
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 AWS PCS computação para usar o modelo de execução.

Exemplo: reserve e use instâncias hpc6a.48xlarge com uma segmentação ODCR

Este exemplo de comando cria um alvo ODCR 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 que seja 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 novoODCR. Para usar o ODCR with AWS PCS, ele deve ser adicionado a um grupo de reserva de capacidade. Isso ocorre porque AWS PCS não suporta indivíduosODCRs. 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 AWS PCS computação 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 AWS PCS computação para usar instâncias hpc6a.48xlarge e use o modelo de execução que faz referência a elas em seu grupo de reserva de capacidade. ODCR 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 uma única ODCR que foi provisionada para um grupo de nós de computação. Mas, AWS PCS suporta muitos outros designs. Por exemplo, você pode subdividir um grupo grande ODCR ou de reserva de capacidade entre vários grupos de nós de computação. Ou você pode usar ODCRs aquela outra AWS conta 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 lançamento 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 (IMDSv2)

O uso da IMDS v2 com EC2 instâncias oferece aprimoramentos significativos de segurança e ajuda a mitigar 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 PCSfilas

Uma AWS PCS fila é uma abstração leve sobre a implementação nativa de uma fila de trabalho do planejador. No caso do Slurm, uma AWS PCS fila 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 AWS PCS cluster 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 em 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 AWS PCS cluster - as filas só podem ser criadas em associação com um PCS cluster específico.
- Um ou mais grupos de nós de AWS PCS computação — uma fila deve estar associada a pelo menos um grupo de nós de PCS computação.

Para criar uma fila em 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 AWS PCS console em `https://console.aws.amazon.com/pcs/home#/clusters`
2. Selecione o cluster em que você deseja criar uma 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 um 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 qualquer tag à sua AWS PCS fila
5. Selecione Criar fila. O campo Status mostrará Criando enquanto a fila está sendo configurada. 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

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

1. Substituir *region-code* com a AWS região na qual você deseja criar seu cluster.
2. Substituir *my-queue* com o 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. Substituir *my-cluster* com o nome ou clusterId do seu cluster.
4. Substitua o valor de `computeNodeGroupId` por seu próprio identificador de grupo de nós de computação. Observe que você não pode especificar nomes de grupos de nós de computação ao criar uma fila.

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

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 AWS PCS fila

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

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 um grupo de nós AWS PCS de computação

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

AWS Management Console

Para atualizar uma fila

1. Abra o AWS PCS console 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. Substituir *region-code* com Região da AWS aquele em que você deseja criar seu cluster.

   b. Substituir *my-queue* com o nome ou `computeNodeGroupId` para sua fila.

   c. Substituir *my-cluster* com o nome ou `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 em AWS PCS

Este tópico fornece uma visão geral de como excluir uma fila em 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 AWS PCSconsole.
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 Slurm.

AWS CLI

Para excluir uma fila

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

AWS PCSnós de login

Um AWS PCS cluster 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 de nós de AWS PCS computação estático configurado para o recurso 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 AWS PCS computação para fornecer nós de login

- Usando instâncias autônomas como nós de AWS PCS login

Usando um grupo de nós de AWS PCS computação 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 AWS PCS computação para fornecer acesso persistente e interativo ao seu cluster.

Criação de um grupo AWS PCS de nós de computação 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 suaVPC, considere usar uma sub-rede pública.
- Se você pretende permitir o SSH acesso, o modelo de lançamento precisará ter um grupo de segurança que exponha a SSH porta aos endereços IP de sua escolha.
- O perfil da IAM instância deve ter somente as AWS permissões que você deseja que seus usuários finais tenham. Para mais detalhes, consulte IAMperfis de instância 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 AWS credenciais 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 o mesmo (ou um derivado) AMI de seus outros grupos de nós de computação para ajudar a garantir que todas as instâncias tenham o mesmo software instalado. Para obter mais informações sobre personalizaçãoAMIs, consulte Amazon Machine Images (AMIs) para AWS PCS

- Configure os mesmos sistemas de arquivos de rede (AmazonEFS, Amazon FSx for Lustre etc.) montados 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 ACTIVE status, 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 em AWS PCS.

Atualização de um grupo de nós de AWS PCS computação 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 AWS PCS de computação.

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. EC2As 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 AWS PCS de computação para nós de login

Você pode atualizar um grupo de nós de login usando o mecanismo de exclusão de grupos de nós de computação em 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 AWS PCS.

Usando instâncias autônomas como nós de AWS PCS login

Você pode configurar EC2 instâncias independentes para interagir com o agendador Slurm de um AWS PCS cluster. Isso é útil para criar nós de login, estações de trabalho ou hosts dedicados de gerenciamento de fluxo de trabalho que funcionam com AWS PCS clusters, mas operam fora do AWS PCS gerenciamento. 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 AWS PCS cluster.
3. Configure adequadamente o Slurm Auth e o Cred Kiosk Daemon (`sackd`) com o endpoint e o segredo do cluster. AWS PCS Para obter mais informações, consulte sackd na documentação do Slurm.

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

Sumário

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

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

- Substituir *`region-code`* com o Região da AWS local em que o cluster de destino está sendo executado.
- Substituir *`cluster-ident`* com o 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": "independent-instance-demo",
        "id": "s3431v9rx2",
        "arn": "arn:aws:pcs:us-east-1:012345678901:cluster/s3431v9rx2",
        "status": "ACTIVE",
```

```
        "createdAt": "2024-07-12T15:32:27.225136+00:00",
        "modifiedAt": "2024-07-12T15:32:27.225136+00:00",
        "scheduler": {
            "type": "SLURM",
            "version": "23.11"
        },
        "size": "SMALL",
        "networking": {
            "subnetIds": [
                "subnet-0123456789abdef"
            ],
            "securityGroupIds": [
                "sg-0123456789abdef"
            ]
        },
        "endpoints": [
            {
                "type": "SLURMCTLD",
                "privateIpAddress": "10.3.149.220",
                "port": "6817"
            }
        ],
        "authKey": {
            "secretArn": "arn:aws:secretsmanager:us-east-1:123456789012:secret:pcs!
slurm-secret-s3431v9rx2-FN7tJFf",
            "secretVersion": "ff58d1fd-070e-4bbc-98a0-64ef967cebcc"
        }
    }
}
```

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 instância do EC2

1. Abra o EC2console 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 um AMI para um dos sistemas operacionais suportados pelo 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 SSH chaves a ser usado para a instância.
7. Na seção Configurações de rede:
   - Selecione a opção Editar.
     i. Selecione o VPC do seu AWS PCS cluster.
     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 AWS PCS cluster de destino. Para obter mais informações, consulte Requisitos e considerações do grupo de segurança.
       B. (Opcional) Selecione um grupo de segurança que permita SSH acesso 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 IAM função 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, substitua *region-code* com a região em que o cluster de destino está sendo executado e substitua *secret-arn* com o valor a ser `secretArn` recuperado na Etapa 1.

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

> **Warning**
>
> 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 somente IMDS aos 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 AWS PCS cluster

Para estabelecer uma conexão com o AWS PCS cluster, execute `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`* com os 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-23.11/sbin/sackd --systemd \$SACKD_OPTIONS
   ExecReload=/bin/kill -HUP \$MAINPID
   KillMode=process
   LimitNOFILE=131072
   LimitMEMLOCK=infinity
   LimitSTACK=infinity

   [Install]
   ```

```
WantedBy=multi-user.target
EOF
```

3. 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 2024-07-16 16:34:55 UTC; 8s ago
 Main PID: 9985 (sackd)
   CGroup: /system.slice/sackd.service
           ##9985 /opt/aws/pcs/scheduler/slurm-23.11/sbin/sackd --systemd --conf-
server=10.3.149.220:6817

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

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-23.11/bin/sinfo
PARTITION AVAIL TIMELIMIT NODES STATE NODELIST
```

```
all up infinite 4 idle~ compute-[1-4]
```

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

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

AWS PCSTrabalho em rede

Seu AWS PCS cluster é criado em uma AmazonVPC. 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 AWS PCS computação. Para obter mais informações sobre modelos de inicialização, consulte Usando modelos de EC2 lançamento da Amazon com AWS PCS.

Tópicos

- AWS PCSVPCe requisitos e considerações de sub-rede
- Criando um VPC para seu AWS PCS cluster
- Grupos de segurança em AWS PCS
- Várias interfaces de rede em AWS PCS
- Grupos de posicionamento para EC2 instâncias em AWS PCS
- Usando o Elastic Fabric Adapter (EFA) com AWS PCS

AWS PCSVPCe requisitos e considerações de sub-rede

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

VPCrequisitos e considerações

Quando você cria um cluster, o VPC que você especifica deve atender aos seguintes requisitos e considerações:

- O 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 VPC do usuário da Amazon.
- Eles VPC devem ter um DNS nome de host e suporte à DNS resolução. Caso contrário, os nós não poderão registrar o cluster do cliente. Para obter mais informações, consulte seus DNS atributos VPC no Guia do VPC usuário da Amazon.
- VPCPode ser necessário usar VPC endpoints AWS PrivateLink para poder entrar em contato com o. AWS PCS API Para obter mais informações, consulte Connect your VPC to services using AWS PrivateLink in Amazon VPC User Guide.

Requisitos e considerações para sub-redes

Quando você cria um cluster Slurm, 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 o clienteVPC. 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 AWS PCS cluster 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 automaticamenteIPv4.
- 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 endereço de rede (NAT) (IPv4), adicione VPC endpoints usando AWS PrivateLink para o cliente. VPC VPCendpoints são necessários para todos os AWS serviços com os quais os nós entram em contato. O único ponto final necessário é AWS PCS permitir que o nó chame a `registerNodeGroupInstances` API ação.
- O status da sub-rede pública ou privada não afeta AWS PCS; os endpoints necessários devem estar acessíveis.

Criando um VPC para seu AWS PCS cluster

Você pode criar uma Amazon Virtual Private Cloud (AmazonVPC) para seus clusters no AWS Parallel Computing Service (AWS PCS).

Use VPC a Amazon para lançar VPC recursos 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 VPC serviço da Amazon antes de implantar VPC clusters de produção. Para obter mais informações, consulte O que é a AmazonVPC? no modo visual do autor. Guia VPC do usuário da Amazon.

Um PCS cluster, nós e recursos de suporte (como sistemas de arquivos e serviços de diretório) são implantados em sua AmazonVPC. Se você quiser usar uma Amazon existente VPC comPCS, ela deverá atender aos requisitos descritos emAWS PCSVPCe requisitos e considerações de sub-rede. Este tópico descreve como criar um VPC que atenda aos PCS requisitos usando um AWS CloudFormation modelo AWS fornecido. 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 paraPCS, você deve ter as IAM permissões necessárias para criar VPC recursos da Amazon. Esses recursos são sub-redesVPCs, grupos de segurança, tabelas e rotas de rotas, internet e NAT gateways. Para obter mais informações, consulte Criar uma VPC com uma

sub-rede pública no Guia do VPC usuário da Amazon. Para revisar a lista completa da AmazonEC2, consulte Ações, recursos e chaves de condição da Amazon EC2 na Referência de autorização de serviço.

Crie uma Amazon VPC

Crie um VPC copiando e colando o apropriado URL para o Região da AWS local onde você PCS usará. 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 seguida VPCCidrBlock, insira `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 um 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 isso AWS CloudFormation pode criar IAM recursos.

Monitore o status da AWS CloudFormation pilha. Quando chegar`CREATE_COMPLETE`, o VPC recurso 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 em 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 AWS PCS computação 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 PCScria uma interface de rede elástica (ENI) entre contas na sub-rede que você especifica ao criar um cluster. Isso fornece ao HPC agendador, que está sendo executado em uma conta gerenciada por AWS, um caminho para se comunicar com as EC2 instâncias iniciadas pelo AWS PCS. Você deve fornecer um grupo de segurança para isso ENI que permita a comunicação bidirecional entre o agendador ENI e suas instâncias de clusterEC2.

Uma maneira simples de fazer isso é criar um grupo de segurança autorreferenciado permissivo que permita 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 — habilitar 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`
- EFAtráfego entre instâncias do grupo de nós. Para obter mais informações, consulte Preparar um grupo EFA de segurança habilitado no Guia do usuário para instâncias Linux
- Qualquer outro tráfego entre nós exigido pela sua carga de trabalho

Várias interfaces de rede em 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 AWS PCS computação 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 seguintes detalhes:

- A sub-rede de cada interface de rede deve ser a mesma que você escolheu ao configurar o grupo de nós de AWS PCS computação que usará o modelo de execução.
- O dispositivo de rede principal, onde a comunicação de rede rotineira, como HTTPS tráfego SSH e tráfego, ocorrerá, 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 SSH segurança 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 em 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 AWS PCS computação para usar o Elastic Fabric Adapter (EFA).

Para criar um grupo de posicionamento de clusters que funcione com 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 seguir URL e carregue-o no 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 execução do grupo de nós de AWS PCS computação.

Usando o Elastic Fabric Adapter (EFA) com AWS PCS

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 AWS PCS cluster EFA envolve a configuração das instâncias do grupo de nós de AWS PCS computação para serem usadas da EFA seguinte maneira.

Sumário

Instale EFA em um AWS PCS -compatível AMI

O AMI usado no grupo de nós de AWS PCS computação deve ter o EFA driver instalado e carregado. Para obter informações sobre como criar um personalizado AMI com o EFA software instalado, consulteImagens personalizadas da Amazon Machine (AMIs) para AWS PCS.

Identifique EFA instâncias habilitadas EC2

Para serem usadosEFA, todos os tipos de instância permitidos para um grupo de AWS PCS computação devem ser compatíveis EFA e ter o mesmo número de vCPUs (e GPUs se apropriado). Para obter uma lista de instâncias EFA habilitadas, consulte o Elastic Fabric Adapter para HPC cargas de trabalho de ML EC2 na Amazon no Guia do usuário do Amazon Elastic Compute Cloud. Você também pode usar o AWS CLI para ver uma lista de tipos de instância compatíveisEFA. Substituir *`region-code`* com o Região da AWS local onde você usa 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
```

Determine quantas interfaces de rede estão disponíveis

Algumas EC2 instâncias têm várias placas de rede. Isso permite que eles tenham váriosEFAs. Para obter mais informações, consulte Várias interfaces de rede em AWS PCS.

Crie um grupo de segurança para apoiar EFA as comunicações

AWS CLI

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

- *region-code*— Especifique Região da AWS onde você usa AWS PCS, comous-east-1.
- *vpc-id*— Especifique o ID do VPC que você usa 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 EFA segurança 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 suporteEFA. Faça o download do modelo a seguir eURL, 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 o VPC local onde você usará AWS PCS.

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

(Opcional) Crie um grupo de colocação

É recomendável executar todas as instâncias usadas EFA em um grupo de posicionamento de cluster para minimizar a distância física entre elas. Recomendamos que você crie um grupo de posicionamento para cada grupo de nós de computação que você EFA usará. Consulte Grupos de posicionamento para EC2 instâncias em AWS PCS para criar um grupo de posicionamento para seu grupo de nós de computação.

Criar ou atualizar um modelo de EC2 lançamento

EFAas interfaces de rede são configuradas no modelo de EC2 lançamento para um grupo de nós de AWS PCS computação. Se houver várias placas de rede, várias EFAs podem ser configuradas. O grupo EFA de segurança 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 EFA interface. Todos EFA precisam do grupo de segurança que habilita EFA o tráfego (`sg-`*`EfaSecGroupId`*). Outros grupos de segurança, especialmente aqueles que lidam com tráfego regularHTTPS, como SSH ou, só precisam ser 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"]
        }
    ]
}
```

Criar ou atualizar o grupo de nós de computação

Crie ou atualize o grupo de nós de AWS PCS computação com instâncias que tenham o mesmo númerovCPUs, a mesma arquitetura de processador e que sejam compatíveis EFA com todas.

Configure o grupo de nós de computação para usá-lo AMI com o EFA software instalado nele e para usar o modelo de execução que configura interfaces de rede EFA habilitadas.

Teste (opcional) EFA

Você pode demonstrar a comunicação EFA habilitada entre dois nós em um grupo de nós de computação executando o `fi_pingpong` programa, que está incluído na EFA instalação do software. Se esse teste for bem-sucedido, é provável que 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=23.11.8
   OS=Linux 5.10.218-208.862.amzn2.aarch64 #1 SMP Tue Jun 4 16:52:10 UTC 2024
   RealMemory=124518 AllocMem=0 FreeMem=110763 Sockets=64 Boards=1
   State=IDLE+CLOUD ThreadsPerCore=1 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A
   Partitions=efa
   BootTime=2024-07-02T19:00:09 SlurmdStartTime=2024-07-08T19:33:25
   LastBusyTime=2024-07-08T19:33:25 ResumeAfterTime=None
   CfgTRES=cpu=64,mem=124518M,billing=64
   AllocTRES=
   CapWatts=n/a
   CurrentWatts=0 AveWatts=0
```

```
   ExtSensorsJoules=n/a ExtSensorsWatts=0 ExtSensorsTemp=n/a
   Reason=Maintain Minimum Number Of Instances [root@2024-07-02T18:59:00]
   InstanceId=i-04927897a9ce3c143 InstanceType=hpc7g.16xlarge

NodeName=hpc7g-2 Arch=aarch64 CoresPerSocket=1
   CPUAlloc=0 CPUEfctv=64 CPUTot=64 CPULoad=0.00
   AvailableFeatures=hpc7g
   ActiveFeatures=hpc7g
   Gres=(null)
   NodeAddr=10.3.132.211 NodeHostName=ip-10-3-132-211 Version=23.11.8
   OS=Linux 5.10.218-208.862.amzn2.aarch64 #1 SMP Tue Jun 4 16:52:10 UTC 2024
   RealMemory=124518 AllocMem=0 FreeMem=110759 Sockets=64 Boards=1
   State=IDLE+CLOUD ThreadsPerCore=1 TmpDisk=0 Weight=1 Owner=N/A MCS_label=N/A
   Partitions=efa
   BootTime=2024-07-02T19:00:09 SlurmdStartTime=2024-07-08T19:33:25
   LastBusyTime=2024-07-08T19:33:25 ResumeAfterTime=None
   CfgTRES=cpu=64,mem=124518M,billing=64
   AllocTRES=
   CapWatts=n/a
   CurrentWatts=0 AveWatts=0
   ExtSensorsJoules=n/a ExtSensorsWatts=0 ExtSensorsTemp=n/a
   Reason=Maintain Minimum Number Of Instances [root@2024-07-02T18:59:00]
   InstanceId=i-0a2c82623cb1393a7 InstanceType=hpc7g.16xlarge
```

Conecte-se a um dos nós (neste caso de exemplo,`hpc7g-1`) usando SSH (ouSSM). Observe que esse é um endereço IP interno, portanto, talvez você precise se conectar a partir de um dos seus nós de login se usarSSH. Lembre-se também de que a instância precisa ser configurada com uma SSH chave 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 EFA habilitado

Como há várias dependências na configuraçãoEFA, 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 seguir eURL, em seguida, carregue-o no CloudFormation console Região da AWS em que você usa 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 AWS PCS computação, 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 VPC onde seu AWS PCS cluster está implantado.
  - Em NodeGroupSubnetId, escolha a sub-rede em seu cluster em VPC que as instâncias EFA habilitadas 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 ao AWS PCSAPI. Muitos clientes escolhem o grupo de segurança padrão do clusterVPC.
  - Em SshSecurityGroupId, forneça o ID de um grupo de segurança que você está usando para permitir SSH acesso de entrada aos nós em seu cluster.
  - Para SshKeyName, selecione o SSH par de chaves para acesso aos 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 AWS PCS.
- Em Capacidades
  - Marque a caixa “Eu reconheço que isso AWS CloudFormation pode criar IAM recursos”.

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 AWS PCS computação, conforme descrito acima emCriar ou atualizar o grupo de nós de computação.

Usando sistemas de arquivos de rede com AWS PCS

Você pode conectar volumes de armazenamento 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 volumes fornecidos pelos AWS serviços. Os volumes incluem Amazon Elastic File System (AmazonEFS), Amazon FSx for NetApp ONTAP, Amazon FSx for Open ZFS, Amazon FSx for Lustre e Amazon File Cache. Você também pode usar volumes autogerenciados, como NFS servidores.

Este tópico aborda considerações e exemplos do uso de sistemas de arquivos em 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 Lustre pacote apropriado 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 volume de armazenamento compartilhado e as instâncias do grupo de nós de computação.
- As regras do grupo de segurança no volume de armazenamento compartilhado e nas instâncias do grupo de nós de computação devem permitir conexões com as portas relevantes.
- Você deve manter um namespace consistente de POSIX usuários e grupos em todos os recursos que acessam os sistemas de arquivos. Caso contrário, trabalhos e processos interativos executados em seu PCS cluster poderão encontrar erros de permissão.
- As montagens do sistema de arquivos são feitas usando modelos de EC2 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 AmazonEFS, Amazon FSx for Lustre, Amazon FSx for Open ZFS e Amazon File Cache. Expanda a seção relevante abaixo para ver um exemplo de cada montagem de rede.

Amazon EFS

Configuração do sistema de arquivos

Crie um sistema de EFS arquivos da Amazon. Certifique-se de que ele tenha um destino de montagem em cada zona de disponibilidade em que você iniciará instâncias de grupos de nós de PCS computação. 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 PCS computação. 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 EFS arquivos da Amazon. Substitua os seguintes valores nesse script pelos seus próprios detalhes:

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

```
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 no VPC local onde você AWS PCS usará. 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 PCS computação. Certifique-se de que o sistema de arquivos esteja associado a um grupo de segurança que permita acesso de entrada e saída a partir das instâncias do grupo de nós de PCS computação. Para obter mais informações sobre grupos de segurança, consulte Controle de acesso ao sistema de arquivos com a 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 AWS PCS sistema)
- (Opcional) *latest* — Qualquer versão do Lustre compatível com 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 Open ZFS

Configuração do sistema de arquivos

Crie um sistema de ZFS arquivos FSx for Open no VPC local em que você usará AWS 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 AWS PCS computação. 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 AWS PCS computação. Para obter mais informações sobre grupos de segurança, consulte Gerenciando o acesso ao sistema de arquivos com a Amazon VPC no Guia ZFS do usuário do FSx for Open.

Modelo de execução

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

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

```
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 no VPC local onde você usará AWS 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 PCS computação. 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 PCS instâncias e o cache de arquivos. Para obter mais informações sobre grupos de segurança, consulte Controle de acesso ao cache com a 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 PCStrabalha com AMIs o que você fornece, oferecendo grande flexibilidade no software e na configuração encontrados nos nós do seu cluster. Se estiver testando AWS PCS, você pode usar uma amostra AMI fornecida e mantida pela AWS. Se você estiver usando AWS PCS na produção, recomendamos que você crie o seu próprioAMIs. Este tópico aborda como descobrir e usar a amostraAMIs, bem como criar e usar sua própria amostra personalizadaAMIs.

Tópicos

- Usando amostras de Amazon Machine Images (AMIs) com AWS PCS
- Imagens personalizadas da Amazon Machine (AMIs) para AWS PCS
- Instaladores de software para criar de forma personalizada AMIs AWS PCS

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

AWSfornece uma amostra AMIs que você pode usar como ponto de partida para trabalhar AWS PCS.

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

Encontre a AWS PCS amostra atual AMIs

AWS Management Console

AWSPCSAMIsAs amostras têm a seguinte convenção de nomenclatura:

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

Valores aceitos

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

Para encontrar AWS PCS uma amostra AMIs

1. Abra o EC2console da Amazon.
2. Navegue até AMIs.
3. Escolha Imagens públicas.
4. Em Localizar AMI por atributo ou tag, pesquise e AMI usando o nome do modelo.

   Exemplos

   - Slurm AMI 23.11 com suporte para Graviton

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

   - Exemplo AMI para instâncias x86

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

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

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

AWS CLI

Você pode encontrar a AWS PCS amostra mais recente AMI com os comandos a seguir. Substituir *region-code* com o Região da AWS local onde você usa AWS PCS, comous-east-1.

- x86_64

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

- Arm 64

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

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

Saiba mais sobre a AWS PCS amostra AMIs

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

Crie seu próprio AMIs compatível com AWS PCS

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

Imagens personalizadas da Amazon Machine (AMIs) para AWS PCS

AWS PCSfoi 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 AWS PCS agente e uma versão compatível do Slurm instalados e configurados corretamente. Você deve usar os AWS instaladores fornecidos para instalar o AWS PCS software em seu dispositivo personalizado. AMI Recomendamos que você use AWS instaladores fornecidos para instalar o Slurm em seu aplicativo personalizado, AMI mas você pode instalar o Slurm sozinho se preferir (não recomendado).

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

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

Tópicos

Etapa 1 — Executar uma instância temporária

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

Para executar uma instância temporária

1. Abra o EC2console 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 Imagens do aplicativo e do sistema operacional, selecione um 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 SSH acesso 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 AWS PCS agente

Instale o agente que configura as instâncias iniciadas pelo AWS PCS para uso com o Slurm.

Para instalar o agente do 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 AWS PCS agente. Os arquivos de instalação são empacotados em um arquivo tarball () `.tar.gz` compactado. Para fazer download da última versão estável, use o seguinte comando: Substituto *`region`* com o Região da AWS local em que você iniciou sua instância temporária, com`us-east-1`.

```
curl https://aws-pcs-repo-region.s3.amazonaws.com/aws-pcs-agent/aws-pcs-agent-
v1.0.0-1.tar.gz -o aws-pcs-agent-v1.0.0-1.tar.gz
```

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 AWS PCS agente.

5. (Opcional) Verifique a autenticidade e a integridade do pacote de AWS PCS software. 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 GPG chave pública AWS PCS e importe-a para o seu chaveiro. Substituto *`region`* com o 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.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 GPG chave.

```
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 AWS PCS agente 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 AWS PCS software. Substituir *region* com o Região da AWS local em que você iniciou sua instância temporária, como `us-east-1`.

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

A saída deve ser semelhante à seguinte:

```
gpg: assuming signed data in './aws-pcs-agent-v1.0.0-1.tar.gz'
gpg: Signature made Thu Aug  8 18:50:19 2024 CEST
gpg:                using RSA key 4BAA531875430EB0739E6D961BA7F0AF6E34C496
gpg: Good signature from "AWS PCS Packages (AWS PCS Packages)" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 1C24 32C1 862F 64D1 F90A  239A 7EEF 030E DDF5 C21C
   Subkey fingerprint: 4BAA 5318 7543 0EB0 739E  6D96 1BA7 F0AF 6E34 C496
```

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 AWS PCS de instalação do software 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-v1.0.0-1.tar.gz && \
    cd aws-pcs-agent
```

7. Instale o software AWS PCS.

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

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

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

   A saída deve ser semelhante à seguinte:

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

Etapa 3 — Instalar o Slurm

Instale uma versão do Slurm que seja compatível com o. AWS PCS

Para instalar o Slurm

1. Conecte-se à mesma instância temporária em que você instalou o AWS PCS software.
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: Substituto *`region`* com o Região da AWS da sua instância temporária, como`us-east-1`.

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

   Você também pode obter a versão mais recente substituindo o número da versão pelo comando anterior (por exemplo:`aws-pcs-slurm-23.11-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 GPG chave pública AWS PCS e importe-a para o seu chaveiro. Substituto *region* com o 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.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 GPG chave.

   ```
   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. Substituir *region* com o Região da AWS local em que você iniciou sua instância temporária, como`us-east-1`.

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

   A saída deve ser semelhante à seguinte:

```
gpg: assuming signed data in './aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz'
gpg: Signature made Thu Aug  8 14:23:38 2024 CEST
gpg:                using RSA key 4BAA531875430EB0739E6D961BA7F0AF6E34C496
gpg: Good signature from "AWS PCS Packages (AWS PCS Packages)" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 1C24 32C1 862F 64D1 F90A  239A 7EEF 030E DDF5 C21C
   Subkey fingerprint: 4BAA 5318 7543 0EB0 739E  6D96 1BA7 F0AF 6E34 C496
```

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

A saída deve ser semelhante à seguinte:

```
SLURM_INSTALL_DATE='Mon Aug 12 12:38:56 UTC 2024'
SLURM_VERSION='23.11.9'
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 um personalizadoAMI, recomendamos que primeiro crie e teste um AMI com apenas o AWS PCS software e o Slurm instalados e, em seguida, adicione incrementalmente seu próprio software e configurações depois de confirmar o sucesso inicial. AWS PCS

Exemplos

- Software Elastic Fabric Adapter (EFA). Para obter mais informações, consulte Comece com EFA e MPI para HPC cargas de trabalho na Amazon EC2 no Guia do usuário do Amazon Elastic Compute Cloud.
- Cliente Amazon Elastic File System (AmazonEFS). Para obter mais informações, consulte Instalação manual do EFS cliente Amazon 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 FSxusuário 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.
- NVIDIADriver,CUDA, eDCGM, para usar os tipos de instância p* ou g*.

Etapa 5 — Crie um AMI compatível com AWS PCS

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

Para criar um AMI a partir da sua instância temporária

1. Abra o EC2console da Amazon.
2. No painel de navegação, escolha Instances (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 Nome da imagem, insira um nome descritivo para o. AMI
    b. (Opcional) Em Descrição da imagem, insira uma breve descrição da finalidade doAMI.

c. Escolha Criar imagem.

5. No painel de navegação, escolha AMIs.
6. Localize o 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 AWS PCS computação.

Etapa 6 — Use o personalizado AMI com um grupo de nós AWS PCS de computação

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

New compute node group

Para usar o personalizado AMI

1. Abra o AWS PCSconsole.
2. No painel de navegação, escolha Clusters.
3. Escolha o cluster em que você usará o personalizado AMI 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 AMIID, pesquise o nome ou ID do personalizado AMI que você deseja usar. Conclua a configuração do grupo de nós de computação e escolha Criar grupo de nós de computação.
5. (Opcional) Confirme as AMI execuções da instância de suporte. 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 em AWS PCS..

   b. Ao ver uma instância ser iniciada e concluir o processo de bootstrap, confirme se ela está usando o esperadoAMI. Para fazer isso, selecione a instância e, em seguida, inspecione o AMIID em Detalhes. Ele deve corresponder ao AMI que você definiu 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 o personalizado AMI

1. Abra o AWS PCSconsole.
2. No painel de navegação, escolha Clusters.
3. Escolha o cluster em que você usará o personalizado AMI e selecione grupos de nós de computação.
4. Selecione o grupo de nós que você deseja configurar e escolha Editar. Em AMIID, pesquise o nome ou ID do personalizado AMI 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 o AMI ID atualizado. As instâncias existentes continuarão usando as antigas AMI até serem AWS PCS substituídas. Para obter mais informações, consulte Atualização de um grupo de nós AWS PCS de computação.
5. (Opcional) Confirme as AMI execuções da instância de suporte. 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 em AWS PCS..

    b. Ao ver uma instância ser iniciada e concluir o processo de bootstrap, confirme se ela está usando o esperadoAMI. Para fazer isso, selecione a instância e, em seguida, inspecione o AMIID em Detalhes. Ele deve corresponder ao AMI que você definiu 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 AMI funciona conforme o esperado AWS PCS, você pode encerrar a instância temporária para parar de incorrer em cobranças por ela.

Para encerrar a instância temporária

1. Abra o EC2console 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 AWS PCS

AWS fornece um arquivo para download que pode instalar o AWS PCS software 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 de forma personalizada AMIs para uso com AWS PCS ou usar seus próprios métodos.

Sumário

AWS PCSinstalador de software

O instalador do AWS PCS software configura uma instância para trabalhar AWS PCS durante o processo de inicialização da instância. Você deve usar os AWS instaladores fornecidos para instalar o AWS PCS software em seu dispositivo personalizado. AMI

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.

O software AWS fornecido instala o seguinte:

- Slurm na versão principal e de manutenção solicitada (atualmente versão 23.11.8) - Licença 2 GPL

  - 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 PMIX suporte e suporte JWT
  - O Slurm é instalado em `/opt/aws/pcs/schedulers/slurm-23.11`
- Open PMIX (versão 4.2.6) — Licença
  - Open PMIX é instalado como um subdiretório do `/opt/aws/pcs/scheduler/`
- libjwt (versão 1.15.3) — Licença -2.0 MPL
  - 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-23.11.service`
- Se eles não existirem, um usuário e um grupo (`slurm:slurm`) do Slurm são criados comUID/GIDof. `401`
- No Amazon Linux 2 e no Rocky Linux 9, a instalação adiciona o EPEL repositório para instalar o software necessário para criar o Slurm ou suas dependências.
- RHEL9Na 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

O AWS PCS software e os instaladores do Slurm oferecem suporte aos seguintes sistemas operacionais:

- Amazon Linux 2
- RedHat Linux corporativo 9
- Rocky Linux 9
- Ubuntu 22.04

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

Tipos de instâncias compatíveis

AWS PCSo software 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

As seguintes versões principais do Slurm são suportadas:

- Fauna 23.11

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 AWS Support.

Por exemplo, o comando a seguir gera a SHA256 soma de verificação para o tarball do Slurm 23.11.9.

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

Resultado do exemplo:

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

A tabela a seguir lista as somas de verificação das versões recentes dos instaladores. Substituir *us-east-1* com o Região da AWS local onde você usa AWS PCS.

Installer (Instalador)	Baixar URL	SHA256soma de verificação
Fauna 23.11.9	`https://aws-pcs-repo-us-east-1.s3.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz`	`1de7d919c8632fe8e2806611bed4fde1005a4fadc795412456e935c7bba2a9b8`
AWS PCSagente 1.0.0	`https://aws-pcs-repo-us-east-1.s3.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.0.0-1.tar.gz`	`d2d3d68d00c685435c38af471d7e2492dde5ce9eb222d7b6ef0042144b134ce0`

Versões do Slurm em AWS PCS

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 PCSinicialmente suporta o Slurm 23.11. Você pode atualizar sua versão principal do Slurm após o lançamento de uma nova versão. AWS PCSfoi projetado para atualizar automaticamente o controlador Slurm com versões de patch.

Quando o SchedMD encerra o suporte para uma versão principal específica, AWS PCS também encerra o suporte para essa versão principal. AWS PCSenvia 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.

Perguntas frequentes sobre as versões do Slurm

Por quanto tempo é AWS PCS compatível com a versão do Slurm?

AWS PCSsegue os ciclos de suporte do SchedMD para as versões principais. AWS PCSsuporta até 3 versões principais a qualquer momento. Depois que o SchedMD lançar uma nova versão principal, AWS PCS retira a versão mais antiga compatível. AWS PCSlança uma nova versão principal do Slurm o mais rápido possível, mas pode haver um atraso entre o lançamento do SchedMD e sua disponibilidade em. AWS PCS

Quando me AWS PCS notifica sobre o fim da vida útil de suporte (EOSL) para as versões do Slurm?

AWS PCSnotifica você várias vezes, em uma cadência predeterminada, antes da data. EOSL

O que devo fazer quando uma versão do Slurm se aproxima? EOSL

Você deve atualizar suas versões do Slurm antes EOSL para ajudar a manter um ambiente seguro e compatível.

Como posso atualizar meus clusters para usar uma nova versão principal do Slurm?

Para atualizar a versão do Slurm, você deve criar um novo cluster. Você também deve fazer o upgrade para o AWS PCS software equivalente em seu AMI e usá-lo para criar os grupos de nós de computação para seu novo cluster.

Como meus clusters receberão os novos lançamentos da versão de patch do Slurm?

AWS PCSfoi projetado para aplicar patches automaticamente para lidar com vulnerabilidades e exposições comuns do Slurm (). CVEs AWS PCSaplica os patches aos controladores de cluster que são executados em contas internas de propriedade do serviço. Você deve usar as AWS PCS API ações AWS Management Console ou para instalar patches nas EC2 instâncias do seu Conta da AWS.

E se eu não atualizar o Slurm até a EOSL data?

AWS PCSfoi projetado para interromper clusters que têm uma versão do Slurm não suportada. Você deve atualizar a versão principal do Slurm do controlador de cluster e o AWS PCS software instalado nos grupos de nós de computação.

Quantas AWS PCS versões do Slurm são compatíveis?

AWS PCSsuporta até 3 versões principais do Slurm a qualquer momento, incluindo a atual e as 2 versões principais anteriores.

Quais atualizações de versão do Slurm devo aplicar?

É altamente recomendável que você use a mesma versão principal em todos os componentes do seu cluster e instale os patches mais recentes assim que eles forem lançados. AMIsPara seus grupos de nós de computação, você deve usar uma versão do software Slurm compatível com a versão Slurm do controlador de cluster. A versão principal do Slurm em sua AMIs deve estar dentro de duas versões da versão principal do Slurm no controlador de cluster. A versão do Slurm instalada nas AMI e nas EC2 instâncias em execução no cluster não pode ser mais recente do que a versão do Slurm no controlador do cluster. Para manter o suporte ao seu cluster, você AMIs deve usar uma versão AWS PCS de software compatível.

E se eu atualizar a versão principal do Slurm, mas usar o software Slurm mais antigo nos meus grupos de nós AMI de computação?

Você deve atualizar o AWS PCS software para a mesma versão para usar a nova funcionalidade do Slurm. Para obter AWS PCS suporte total, todos os componentes do Slurm devem usar versões compatíveis. Em resumo:

- Somos capazes de fornecer suporte completo quando o controlador de cluster e todos os componentes (AWS PCSpacotes) em seus Conta da AWS dois usam as versões suportadas.
- AWS PCSfoi projetado para interromper um cluster se a versão Slurm de seu controlador chegar. EOSL

- Se a versão Slurm dos componentes estiver Conta da AWS ao seu alcanceEOSL, seu cluster não será suportado.

Em que ordem devo atualizar os componentes no meu cluster?

Você deve atualizar a versão do Slurm do seu controlador de cluster antes de usar um AMI com uma versão mais recente do Slurm. Você atualiza um grupo de nós de computação para usar o. AMI AWS PCSusa o AMI para iniciar novas EC2 instâncias no grupo de nós de computação. AWS PCSnão atualiza EC2 instâncias existentes que têm trabalhos em execução; foi AWS PCS projetado para encerrar essas instâncias após a conclusão dos trabalhos.

AWS PCSOferece suporte estendido para as versões do Slurm?

Não. Comunicaremos informações detalhadas sobre as opções de suporte estendido, incluindo quaisquer custos adicionais e a cobertura de suporte específica fornecida.

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 AWS PCS. Os tópicos a seguir mostram como configurar para atender AWS PCS aos seus objetivos de segurança e conformidade. Você também aprenderá a usar outros AWS serviços que ajudam a monitorar e proteger seus AWS PCS recursos.

Tópicos

- Melhores práticas de segurança para serviços de computação AWS paralela

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 seu 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 privacidade de dados, consulte Privacidade de dados FAQ. Para obter informações sobre proteção de dados na Europa, consulte o Modelo de Responsabilidade AWS Compartilhada e GDPR a postagem no blog AWS de segurança.

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 a autenticação multifator (MFA) com cada conta.
- UseSSL/TLSpara se comunicar com AWS os recursos. Exigimos TLS 1,2 e recomendamos TLS 1,3.
- Configure API e registre as atividades do usuário com AWS CloudTrail.
- 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 FIPS 140-3 módulos criptográficos validados ao acessar AWS por meio de uma interface de linha de comando ou umaAPI, use um endpoint. FIPS Para obter mais informações sobre os FIPS endpoints disponíveis, consulte Federal Information Processing Standard (FIPS) 140-3.

É altamente recomendável que nunca sejam colocadas informações de identificação confidenciais, como endereços de e-mail dos seus clientes, em marcações ou campos de formato livre, como um campo Nome. Isso inclui quando você trabalha com AWS PCS ou Serviços da AWS usa o

console,API, AWS CLI, ou AWS SDKs. Quaisquer dados inseridos em tags ou 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, é altamente recomendável que você não inclua informações de credenciais no URL para validar sua solicitação para esse servidor.

Criptografia em repouso

A criptografia é ativada por padrão para dados em repouso quando você cria um cluster do Serviço de Computação AWS Paralela (AWS PCS) com o AWS Management Console AWS CLI, AWS PCSAPI,, ou AWS SDKs. AWS PCSusa uma KMSchave AWS própria para criptografar dados em repouso. Para obter mais informações, consulte Chaves do cliente e AWS chaves no Guia do AWS KMS desenvolvedor. O segredo do cluster é armazenado AWS Secrets Manager e criptografado com a KMS chave gerenciada do Secrets Manager. Para obter mais informações, consulte Trabalhando com segredos de cluster em AWS PCS.

Em um AWS PCS cluster, 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 PCSexclui os dados do trabalho após a conclusão do trabalho.
- Segredo de autenticação do agendador — o AWS PCS usa para autenticar todas as comunicações do agendador no cluster.

Para obter informações sobre o estado do agendador, criptografa AWS PCS 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 o AWS PCS API usam TLS criptografia 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 Assinar AWS API solicitações no Guia AWS Identity and Access Management do usuário. AWS gerencia o API controle de acesso por meio das IAM políticas das credenciais de segurança que você usa para se conectar.

AWS PCSusa 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 entre as instâncias do cluster permanecem locais EC2 VPC e, portanto, estão sujeitos à VPC criptografia 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 PCSusa uma AWS KMSchave própria para criptografar dados. Para obter mais informações, consulte Chaves do cliente e AWS chaves no Guia do AWS KMS desenvolvedor. O segredo do cluster é armazenado AWS Secrets Manager e criptografado com a KMS chave gerenciada do Secrets Manager. Para obter mais informações, consulte Trabalhando com segredos de cluster em AWS PCS.

Privacidade do tráfego entre redes

AWS PCSos recursos computacionais de um cluster residem VPC em 1 na conta do cliente. Portanto, todo o tráfego de AWS PCS serviço interno em um cluster permanece na AWS rede e não viaja pela Internet. A comunicação entre o usuário e AWS PCS os nós pode viajar pela Internet e recomendamos usar SSH nosso Systems Manager para se conectar 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 o AWS PCS API para realizar tarefas administrativas para o serviço. Você e seus usuários acessam as portas do endpoint do Slurm para interagir diretamente com o agendador.

Criptografando o tráfego API

Para acessar o AWS PCSAPI, 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

oferecer suporte a pacotes de criptografia com Perfect Forward Secrecy (PFS), como Ephemeral Diffie-Hellman () ou Elliptic Curve Diffie-Hellman Ephemeral (). DHE ECDHE A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos. Além disso, as solicitações devem ser assinadas usando uma ID de chave de acesso e uma chave de acesso secreta associada a um IAM principal. 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.

Acesse o serviço de computação AWS paralela usando um endpoint de interface ()AWS PrivateLink

Você pode usar AWS PrivateLink para criar uma conexão privada entre você VPC e o Serviço de Computação AWS Paralela (AWS PCS). Você pode acessar AWS PCS como se estivesse no seuVPC, sem o uso de um gateway de internet, NAT dispositivo, VPN conexão ou AWS Direct Connect conexão. As instâncias em seu VPC não precisam de endereços IP públicos para serem acessadas AWS PCS.

Você estabelece essa conectividade privada criando um endpoint de interface, desenvolvido pelo AWS PrivateLink. Criaremos um endpoint de interface de rede em cada sub-rede que você habilitar para o endpoint de interface. Essas são interfaces de rede gerenciadas pelo solicitante que servem como ponto de entrada para o tráfego destinado a. 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 Acessar um AWS serviço usando um VPC endpoint de interface no AWS PrivateLink Guia.

AWS PCSsuporta fazer chamadas para todas as suas API ações por meio do endpoint da interface.

Se você VPC não tiver acesso direto à Internet, deverá configurar um VPC endpoint para permitir que suas instâncias do grupo de nós de computação chamem a AWS PCS `RegisterComputeNodeGroupInstance`APIação.

Crie um endpoint de interface para AWS PCS

Você pode criar um endpoint de interface para AWS PCS usar o VPC console da Amazon ou o AWS Command Line Interface (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
```

Substituir *`region`* com o ID do Região da AWS para criar o endpoint, como`us-east-1`.

Se você habilitar private DNS para o endpoint da interface, poderá fazer API solicitações AWS PCS usando seu DNS nome 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 IAM recurso que você pode anexar a um endpoint de interface. A política de endpoint padrão permite acesso total AWS PCS por meio do endpoint da interface. Para controlar o acesso permitido pelo seuVPC, anexe uma política AWS PCS de endpoint personalizada ao endpoint da interface.

Uma política de endpoint especifica as seguintes informações:

- Os diretores que podem realizar ações (Contas da AWS, IAM usuários e IAM funções).
- As ações que podem ser executadas.
- Os recursos nos quais as ações podem ser executadas.

Para obter mais informações, consulte Controlar o Acesso a Serviços Usando Políticas de Endpoint no AWS PrivateLink Guia.

Exemplo: política VPC de endpoint para AWS PCS ações

O exemplo a seguir refere-se a uma política de endpoint personalizada. 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*. Substituir *region* com o ID Região da AWS do cluster, comous-east-1. Substituir *account-id* com o 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. IAMos administradores controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) a usar AWS PCS os recursos. IAMé um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Tópicos

- Funções vinculadas ao serviço para o AWS PCS
- Função do Amazon EC2 Spot para AWS PCS
- Permissões mínimas para AWS PCS
- IAMperfis de instância para o AWS Parallel Computing Service
- Solução de problemas de identidade e acesso ao serviço de computação AWS paralela

Público

A forma como você usa AWS Identity and Access Management (IAM) difere, dependendo do trabalho que você faz AWS PCS.

Usuário do serviço — Se você usar o AWS PCS serviço para realizar seu trabalho, seu administrador fornecerá as credenciais e as permissões de que você precisa. À medida que você usa mais AWS PCS recursos para fazer seu trabalho, talvez precise de permissões adicionais. Entender como o acesso é gerenciado pode ajudar você a solicitar as permissões corretas ao seu 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 AWS PCS recursos da sua empresa, provavelmente tem acesso total AWS PCS a. É seu trabalho determinar quais AWS PCS recursos e recursos seus usuários do serviço devem acessar. Em seguida, você deve enviar solicitações ao IAM administrador para alterar as permissões dos usuários do serviço. Revise as informações nesta página para entender os conceitos básicos doIAM. Para saber mais sobre como sua empresa pode usar IAM com AWS PCS, consulteComo o serviço de computação AWS paralela funciona com IAM.

IAMadministrador — Se você for IAM administrador, talvez queira saber detalhes sobre como criar políticas para gerenciar o acesso AWS PCS. Para ver exemplos de políticas AWS PCS baseadas em identidade que você pode usar emIAM, consulte. Exemplos de políticas baseadas em identidade para o serviço de computação AWS paralela

Autenticando com identidades

A autenticação é a forma como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado (conectado AWS) como IAM usuário ou assumindo uma IAM função. Usuário raiz da conta da AWS

Você pode entrar AWS como uma identidade federada usando credenciais fornecidas por meio de uma fonte de identidade. AWS IAM Identity Center Os usuários (do 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 entra como uma identidade federada, seu administrador configurou previamente a federação de identidades usando IAM funções. 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 em 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 você mesmo assinar solicitações, consulte Assinar AWS API solicitações no Guia IAM do usuário.

Independente do método de autenticação usado, também pode ser exigido que você forneça 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 AWS IAM Identity Center do usuário e Uso da autenticação multifator (MFA) AWS no Guia do IAMusuário.

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 ver a lista completa de tarefas que exigem que você faça login como usuário raiz, consulte Tarefas que exigem credenciais de usuário raiz no Guia do IAM usuário.

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, recomendamos 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 informações sobre o IAM Identity Center, consulte O que é o IAM Identity Center? no Guia do AWS IAM Identity Center usuário.

Grupos e usuários do IAM

Um IAMusuário é uma identidade dentro da sua Conta da AWS que tem permissões específicas para uma única pessoa ou aplicativo. Sempre que possível, recomendamos confiar em credenciais temporárias em vez de criar IAM usuários que tenham 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 IAM os usuários, recomendamos que você alterne as chaves de acesso. Para obter mais informações, consulte Alterne as chaves de acesso regularmente para casos de uso que exigem credenciais de longo prazo no Guia do IAMusuário.

Um IAMgrupo é uma identidade que especifica uma coleção de IAM usuários. 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 IAM recursos.

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 Quando criar um IAM usuário (em vez de uma função) no Guia do IAM usuário.

IAMfunções

Uma IAMfunção é uma identidade dentro da sua Conta da AWS que tem permissões específicas. É semelhante a um IAM usuário, mas não está associado a uma pessoa específica. Você pode assumir temporariamente uma IAM função no AWS Management Console trocando de funções. Você pode assumir uma função chamando uma AWS API operação AWS CLI or ou usando uma personalizadaURL. Para obter mais informações sobre métodos de uso de funções, consulte Usando IAM funções no Guia IAM do usuário.

IAMfunções com credenciais temporárias são úteis nas seguintes situações:

- Acesso de usuário federado: para atribuir permissões a identidades federadas, você pode 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 pelo mesmo. Para obter informações sobre funções para federação, consulte Criação de uma função para um provedor de identidade terceirizado no Guia IAM do usuário. Se você usa o IAM Identity Center, configura um conjunto de permissões. Para controlar o que suas identidades podem acessar após a autenticação, o IAM Identity Center correlaciona o conjunto de permissões a uma função em. IAM Para obter informações sobre conjuntos de permissões, consulte Conjuntos de Permissões no Manual do Usuário do AWS IAM Identity Center .
- Permissões temporárias IAM de IAM usuário — Um usuário ou função pode assumir uma IAM função para assumir temporariamente permissões diferentes para uma tarefa específica.
- Acesso entre contas — Você pode usar uma IAM função para permitir que alguém (um diretor confiável) em uma conta diferente 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 saber a diferença entre funções e políticas baseadas em recursos para acesso entre contas, consulte Acesso a recursos entre contas IAM no Guia do IAM usuário.
- 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 do principal de chamada, usando um perfil de serviço ou um perfil vinculado a serviço.
  - Sessões de acesso direto (FAS) — Quando você usa um IAM usuário ou uma função para realizar ações em AWS, você é considerado principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. FASusa 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. FASas solicitações 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 FAS solicitações, consulte Encaminhar sessões de acesso.
  - Função de serviço — Uma função de serviço é uma IAMfunção que um serviço assume para realizar ações em seu nome. Um IAM administrador pode criar, modificar e excluir uma função de serviço internamenteIAM. Para obter mais informações, consulte Criação de uma função para delegar permissões a uma AWS service (Serviço da AWS) no Guia do IAM usuário.

  - 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 a função de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um IAM administrador pode visualizar, mas não editar, as permissões das funções vinculadas ao serviço.
- Aplicativos em execução na Amazon EC2 — Você pode usar uma IAM função para gerenciar credenciais temporárias para aplicativos que estão sendo executados em uma EC2 instância e fazendo AWS CLI AWS API solicitações. 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 os programas em execução na EC2 instância recebam credenciais temporárias. Para obter mais informações, consulte Como usar uma IAM função para conceder permissões a aplicativos executados em EC2 instâncias da Amazon no Guia IAM do usuário.

Para saber se usar IAM funções ou IAM usuários, consulte Quando criar uma IAM função (em vez de um usuário) no Guia do IAM usuário.

Gerenciando 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 JSON documentos. Para obter mais informações sobre a estrutura e o conteúdo dos documentos de JSON política, consulte Visão geral das JSON políticas no Guia IAM do usuário.

Os administradores podem usar AWS JSON políticas 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 funções não têm permissões. Para conceder permissão aos usuários para realizar ações nos recursos de que precisam, um IAM administrador pode criar IAM políticas. O administrador pode então adicionar as IAM políticas às funções e os usuários podem assumir as funções.

IAMas políticas definem permissões para uma ação, independentemente do método usado para realizar 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 do AWS Management Console AWS CLI, do ou do AWS API.

Políticas baseadas em identidade

Políticas baseadas em identidade são documentos de políticas de JSON permissões que você pode anexar a uma identidade, como um IAM usuário, grupo de usuários ou função. 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 Criação de IAM políticas no Guia do IAMusuário.

As políticas baseadas em identidade podem ser categorizadas ainda adicionalmente 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 Escolha entre políticas gerenciadas e políticas em linha no Guia do IAMusuário.

Políticas baseadas no recurso

Políticas baseadas em recursos são documentos JSON de política que você anexa a um recurso. Exemplos de políticas baseadas em recursos são as políticas de confiança de IAM funções e as políticas de bucket do Amazon S3. Em serviços que suportem políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações um principal especificado pode executar nesse recurso 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 de uma política baseada IAM 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. ACLssão semelhantes às políticas baseadas em recursos, embora não usem o formato de documento JSON de política.

Amazon S3, AWS WAF, e Amazon VPC são exemplos de serviços que oferecem suporte. ACLs Para saber maisACLs, 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 as permissões máximas que uma política baseada em identidade pode conceder a uma IAM entidade (IAMusuário ou função). 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 IAM entidades no Guia IAM do usuário.
- Políticas de controle de serviço (SCPs) — SCPs são JSON políticas 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. Os SCP limites de permissões para entidades nas contas dos membros, incluindo cada uma Usuário raiz da conta da AWS. Para obter mais informações sobre Organizations eSCPs, consulte Políticas de controle de serviços 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 atributo. Uma negação explícita em qualquer uma dessas políticas substitui a permissão. Para obter mais informações, consulte Políticas de sessão no Guia IAM do usuário.

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 política estão envolvidos, consulte Lógica de avaliação de políticas no Guia IAM do usuário.

Como o serviço de computação AWS paralela funciona com IAM

Antes de usar IAM para gerenciar o acesso ao AWS PCS, saiba quais IAM recursos estão disponíveis para uso AWS PCS.

IAMrecursos que você pode usar com o AWS Parallel Computing Service

IAMrecurso	AWS PCSapoio
Políticas baseadas em identidade	Sim
Políticas baseadas em recursos	Não
Ações das políticas	Sim
Atributos de políticas	Sim
Chaves de condição de política (específicas do serviço)	Sim
ACLs	Não
ABAC(tags nas políticas)	Sim
Credenciais temporárias	Sim
Permissões de entidade principal	Sim
Perfis de serviço	Não
Funções vinculadas ao serviço	Sim

Para obter uma visão geral de como AWS PCS e outros AWS serviços funcionam com a maioria dos IAM recursos, consulte AWS os serviços que funcionam com IAM no Guia do IAM usuário.

Políticas baseadas em identidade para AWS PCS

Compatível com políticas baseadas em identidade: Sim

Políticas baseadas em identidade são documentos de políticas de JSON permissões que você pode anexar a uma identidade, como um IAM usuário, grupo de usuários ou função. 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 Criação de IAM políticas no Guia do IAMusuário.

Com políticas IAM baseadas em identidade, você pode especificar ações e recursos permitidos ou negados, bem 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 você pode usar em uma JSON política, consulte a referência IAM JSON de elementos de política no Guia IAM do usuário.

Exemplos de políticas baseadas em identidade para AWS PCS

Para ver exemplos de políticas AWS PCS baseadas em identidade, consulte. Exemplos de políticas baseadas em identidade para o serviço de computação AWS paralela

Políticas baseadas em recursos dentro AWS PCS

Suporte a políticas baseadas em recursos: não

Políticas baseadas em recursos são documentos JSON de política que você anexa a um recurso. Exemplos de políticas baseadas em recursos são as políticas de confiança de IAM funções e as políticas de bucket do Amazon S3. Em serviços que suportem políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o recurso ao qual a política está anexada, a política define quais ações um principal especificado pode executar nesse recurso 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 habilitar o acesso entre contas, você pode especificar uma conta ou IAM entidades inteiras em outra conta como 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 IAM administrador 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. Para obter mais informações, consulte Acesso a recursos entre contas IAM no Guia do IAM usuário.

Ações políticas para AWS PCS

Compatível com ações de políticas: Sim

Os administradores podem usar AWS JSON políticas para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O `Action` elemento de uma JSON política descreve as ações que você pode usar para permitir ou negar acesso em uma política. As ações de política geralmente têm o mesmo nome da AWS API operação associada. Há algumas exceções, como ações somente de permissão que não têm uma operação correspondente. API 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 AWS PCS ações, consulte Ações definidas pelo serviço de computação AWS paralela na Referência de autorização de serviço.

As ações de política AWS PCS usam o seguinte prefixo antes da ação:

```
pcs
```

Para especificar várias ações em uma única instrução, separe-as com vírgulas.

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

Para ver exemplos de políticas AWS PCS baseadas em identidade, consulte. Exemplos de políticas baseadas em identidade para o serviço de computação AWS paralela

Recursos políticos para AWS PCS

Compatível com recursos de políticas: Sim

Os administradores podem usar AWS JSON políticas para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos, e em que condições.

O elemento `Resource` JSON de política especifica o objeto ou objetos aos quais a ação se aplica. As instruções devem incluir um elemento `Resource` ou `NotResource`. Como prática recomendada, especifique um recurso usando seu Amazon Resource Name (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 AWS PCS recursos e seusARNs, consulte Recursos definidos pelo serviço de computação AWS paralela na Referência de autorização de serviço. Para saber com quais ações você pode especificar cada recurso, consulte Ações definidas pelo serviço de computação AWS paralela. ARN

Para ver exemplos de políticas AWS PCS baseadas em identidade, 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 AWS JSON políticas 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 `Condition` em uma instrução ou várias chaves em um único `Condition` elemento, 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, você pode conceder permissão a um IAM usuário para acessar um recurso somente se ele estiver marcado com o nome de IAM usuário. Para obter mais informações, consulte elementos de IAM política: variáveis e tags no Guia IAM do usuário.

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 IAM do usuário.

Para ver uma lista de chaves de AWS PCS condição, 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 AWS PCS baseadas em identidade, consulte. Exemplos de políticas baseadas em identidade para o serviço de computação AWS paralela

ACLsem AWS PCS

SuportesACLs: Não

As listas de controle de acesso (ACLs) controlam quais diretores (membros da conta, usuários ou funções) têm permissões para acessar um recurso. ACLssão semelhantes às políticas baseadas em recursos, embora não usem o formato de documento JSON de política.

ABACcom AWS PCS

Suportes ABAC (tags nas políticas): Sim

O controle de acesso baseado em atributos (ABAC) é uma estratégia de autorização que define permissões com base em atributos. Em AWS, esses atributos são chamados de tags. Você pode anexar tags a IAM entidades (usuários ou funções) e a muitos AWS recursos. Marcar entidades e

recursos é a primeira etapa doABAC. Em seguida, você cria ABAC políticas para permitir operações quando a tag do diretor corresponde à tag do recurso que ele está tentando acessar.

ABACé útil em ambientes que estão crescendo rapidamente e ajuda em situações em que o gerenciamento de políticas se torna complicado.

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 oferecer suporte às três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço oferecer suporte às três chaves de condição somente para alguns tipos de recursos, o valor será Parcial

Para obter mais informações sobreABAC, consulte O que éABAC? no Guia do IAM usuário. Para ver um tutorial com etapas de configuraçãoABAC, consulte Usar controle de acesso baseado em atributos (ABAC) no Guia do IAMusuário.

Usando credenciais temporárias com AWS 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 esse trabalho IAM no Guia do IAM usuário.

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 a troca de funções, consulte Alternando para uma função (console) no Guia IAM do usuário.

Você pode criar manualmente credenciais temporárias usando o AWS CLI ou AWS API. 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 emIAM.

Permissões principais entre serviços para AWS PCS

Suporta sessões de acesso direto (FAS): Sim

Quando você usa um IAM usuário ou uma função para realizar ações em AWS, você é considerado principal. Ao usar alguns serviços, você pode executar uma ação que inicia outra ação em um serviço diferente. FASusa 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. FASas solicitações 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 FAS solicitações, consulte Encaminhar sessões de acesso.

Funções de serviço para AWS PCS

Compatível com perfis de serviço: não

Uma função de serviço é uma IAMfunção que um serviço assume para realizar ações em seu nome. Um IAM administrador pode criar, modificar e excluir uma função de serviço internamenteIAM. Para obter mais informações, consulte Criação de uma função para delegar permissões a uma AWS service (Serviço da AWS) no Guia do IAM usuário.

> **Warning**
>
> Alterar as permissões de uma função de serviço pode interromper AWS PCS a funcionalidade. Edite as funções de serviço somente quando AWS PCS fornecer orientação para fazer isso.

Funções vinculadas a serviços para AWS PCS

Suporte a 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 a função de executar uma ação em seu nome. As funções vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um IAM administrador pode visualizar, mas não editar, as permissões das funções vinculadas ao serviço.

Para obter detalhes sobre como criar ou gerenciar funções vinculadas a serviços, consulte AWS serviços que funcionam com. IAM Encontre um serviço na tabela que inclua um `Yes` na coluna Função vinculada ao serviço. Escolha o link Sim para visualizar a documentação do perfil vinculado a serviço desse serviço.

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 AWS PCS recursos. Eles também não podem realizar tarefas usando o AWS Management Console, AWS Command Line Interface (AWS CLI) ou AWS API. Para conceder permissão aos usuários para realizar ações nos recursos de que precisam, um IAM administrador pode criar IAM políticas. O administrador pode então adicionar as IAM políticas às funções e os usuários podem assumir as funções.

Para saber como criar uma política IAM baseada em identidade usando esses exemplos de documentos de JSON política, consulte Criação de IAM políticas no Guia do IAMusuário.

Para obter detalhes sobre ações e tipos de recursos definidos por 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

- Melhores práticas de política
- Usando o AWS PCS console
- Permitir que usuários visualizem suas próprias permissões

Melhores práticas de política

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir AWS PCS recursos em sua conta. Essas ações podem incorrer em custos para seus 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 AWS gerenciadas ou políticas AWS gerenciadas para funções de trabalho no Guia IAM do usuário.
- Aplique permissões com privilégios mínimos — Ao definir permissões com IAM políticas, conceda somente as permissões necessárias para realizar uma tarefa. Você faz isso definindo as ações que podem ser executadas em atributos específicos sob condições específicas, também

conhecidas como permissões de privilégio mínimo. Para obter mais informações sobre IAM como usar para aplicar permissões, consulte Políticas e permissões IAM no Guia IAM do usuário.

- Use condições nas IAM políticas para restringir ainda mais o acesso — Você pode adicionar uma condição às suas 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 usandoSSL. 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 IAM JSON da política: Condição no Guia IAM do usuário.
- Use o IAM Access Analyzer para validar suas IAM políticas e garantir permissões seguras e funcionais — o IAM Access Analyzer valida políticas novas e existentes para que as políticas sigam a linguagem da IAM política (JSON) e as melhores práticas. IAM IAMO Access Analyzer fornece mais de 100 verificações de políticas e recomendações práticas para ajudá-lo a criar políticas seguras e funcionais. Para obter mais informações, consulte Validação da política do IAM Access Analyzer no Guia do IAM Usuário.
- Exigir autenticação multifatorial (MFA) — Se você tiver um cenário que exija IAM usuários ou um usuário root Conta da AWS, ative MFA para obter segurança adicional. Para exigir MFA quando API as operações são chamadas, adicione MFA condições às suas políticas. Para obter mais informações, consulte Configurando o API acesso MFA protegido no Guia do IAMusuário.

Para obter mais informações sobre as melhores práticas emIAM, consulte as melhores práticas de segurança IAM no Guia IAM do usuário.

Usando o AWS PCS console

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 AWS PCS recursos em seu Conta da AWS. Se você criar uma política baseada em identidade que seja mais restritiva do 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 AWS CLI o. ou AWS API o. Em vez disso, permita o acesso somente às ações que correspondam à API operação que eles estão tentando realizar.

Para obter mais informações sobre as permissões mínimas necessárias para usar o AWS PCS console, consultePermissões mínimas para AWS PCS.

Permitir que usuários visualizem suas próprias permissões

Este exemplo mostra como você pode criar uma política que permita IAM aos usuários visualizar as políticas embutidas e gerenciadas que estão anexadas à identidade do usuário. Essa política inclui permissões para concluir essa ação no console ou programaticamente usando o AWS CLI ou. AWS API

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS 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 políticas gerenciadas pelo cliente da 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 API operações são disponibilizadas para os serviços existentes.

Para obter mais informações, consulte as políticas AWS gerenciadas no Guia IAM do usuário.

AWS política gerenciada: AWSPCSServiceRolePolicy

Você não pode se vincular AWSPCSServiceRolePolicy às suas IAM entidades. Essa política está vinculada a uma função vinculada ao serviço que permite AWS PCS realizar ações em seu nome. Para obter mais informações, consulte Funções vinculadas ao serviço para o AWS PCS.

Detalhes das permissões

Esta política inclui as seguintes permissões:

- `ec2`— Permite AWS PCS criar e gerenciar EC2 recursos da Amazon.
- `iam`— Permite AWS PCS criar uma função vinculada a serviços para a EC2 frota da Amazon e passar a função para a Amazon. EC2
- `cloudwatch`— Permite AWS PCS publicar métricas de serviço na Amazon CloudWatch.
- `secretsmanager`— Permite AWS PCS gerenciar segredos para recursos de AWS PCS cluster.

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

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

```
            }
        },
        {
            "Sid": "PermissionsToTerminatePCSManagedInstances",
            "Effect": "Allow",
            "Action": [
                "ec2:TerminateInstances"
            ],
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "Null": {
                    "aws:ResourceTag/AWSPCSManaged": "false"
                }
            }
        },
        {
            "Sid": "PermissionsToPassRoleToEC2",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": [
                "arn:aws:iam::*:role/*/AWSPCS*",
                "arn:aws:iam::*:role/AWSPCS*",
                "arn:aws:iam::*:role/aws-pcs/*",
                "arn:aws:iam::*:role/*/aws-pcs/*"
            ],
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": [
                        "ec2.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Sid": "PermissionsToControlClusterInstanceAttributes",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances",
                "ec2:CreateFleet"
            ],
            "Resource": [
                "arn:aws:ec2:*::image/*",
                "arn:aws:ec2:*::snapshot/*",
                "arn:aws:ec2:*:*:subnet/*",
```

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

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

AWS PCSatualizações nas políticas AWS gerenciadas

Veja detalhes sobre as atualizações das políticas AWS gerenciadas AWS PCS desde que esse serviço começou a rastrear essas alterações. Para receber alertas automáticos sobre alterações nessa página, assine o RSS feed na página Histórico do AWS PCS documento.

Alteração	Descrição	Data
O AWS PCS iniciou o rastreamento das alterações	AWS PCScomeçou a rastrear as mudanças em suas políticas AWS gerenciadas.	28 de agosto de 2024

Funções vinculadas ao serviço para o AWS PCS

AWS O Serviço de Computação Paralela usa AWS Identity and Access Management (IAM) funções vinculadas ao serviço. Uma função vinculada ao serviço é um tipo exclusivo de IAM função vinculada diretamente a. AWS PCS As funções vinculadas ao serviço são predefinidas 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 AWS PCS porque você não precisa adicionar manualmente as permissões necessárias. AWS PCSdefine as permissões de suas funções vinculadas ao serviço e, a menos que seja definido de outra forma, só AWS PCS pode assumir suas funções. As permissões definidas incluem a política de confiança e a política de permissões, e essa política de permissões não pode ser anexada a nenhuma outra IAM entidade.

Você só pode excluir um perfil vinculado a serviço depois de excluir os recursos relacionados. Isso protege seus AWS PCS recursos 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 da função vinculada a esse serviço.

Permissões de função vinculadas ao serviço para AWS PCS

AWS PCSusa a função vinculada ao serviço chamada AWSServiceRoleForPCS— Permitir AWS PCS gerenciar recursos da AmazonEC2.

A função AWSServiceRoleForPCS vinculada ao serviço confia nos seguintes serviços para assumir a função:

- `pcs.amazonaws.com`

A política de permissões de função nomeada AWSPCSServiceRolePolicy AWS PCSpermite concluir 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 Permissões de funções vinculadas ao serviço no Guia do IAMusuário.

Criação de uma função vinculada ao serviço para AWS PCS

Você não precisa criar manualmente uma função vinculada ao serviço. AWS PCScria uma função vinculada ao serviço para você ao criar um cluster.

Editando uma função vinculada ao serviço para AWS PCS

AWS PCSnão permite que você edite a função AWSServiceRoleForPCS vinculada ao serviço. 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, você pode editar a descrição da função usandoIAM. Para obter mais informações, consulte Edição de uma função vinculada ao serviço no Guia do IAMusuário.

Excluindo uma função vinculada ao serviço para AWS PCS

Se você não precisar mais usar um atributo ou serviço que requer uma função vinculada a serviço, é recomendável excluí-la. Dessa forma, você não tem uma entidade não utilizada que não seja monitorada ativamente ou mantida. No entanto, você deve limpar os recursos de sua função vinculada ao serviço antes de excluí-la manualmente.

> **Note**
>
> Se o AWS PCS serviço 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 AWS PCS recursos usados pelo AWSServiceRoleForPCS

Você deve excluir todos os seus clusters para excluir a função AWSServiceRoleForPCS vinculada ao serviço. Para obter mais informações, consulte Excluir um cluster.

Para excluir manualmente a função vinculada ao serviço usando IAM

Use o IAM console AWS CLI, o ou o AWS API para excluir a função AWSServiceRoleForPCS vinculada ao serviço. Para obter mais informações, consulte Excluindo uma função vinculada ao serviço no Guia do IAM usuário.

Regiões suportadas por funções vinculadas ao serviço do AWS PCS

AWS PCSsuporta o uso de funções vinculadas ao serviço 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 AWS PCS computação que use o Spot como opção de compra, você também deve ter a função AWSServiceRoleForEC2Spotvinculada ao serviço 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 `AWSServiceRoleForEC2Spot` IAM função.
>
> ```
> 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 IAM permissões mínimas necessárias para que uma IAM identidade (usuário, grupo ou função) use o serviço.

Sumário

- Permissões mínimas necessárias para usar tags
- Permissões mínimas necessárias para suportar registros
- Permissões mínimas para um administrador de serviços

Permissões mínimas para usar API ações

APIação	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`	
ListClusters	`pcs:ListClusters`	
GetCluster	`pcs:GetCluster`	`ec2:DescribeSubnets`
DeleteCluster	`pcs:DeleteCluster`	
CreateComputeNodeGroup	`ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:DescribeLaunchTemplates,`	`iam:ListInstanceProfiles, ec2:DescribeImages, pcs:GetCluster`

APIação	Permissões mínimas	Permissões adicionais para o console
	`ec2:DescribeLaunchTemplateVersions`, `ec2:DescribeInstanceTypes`, `ec2:RunInstances`, `ec2:CreateFleet`, `ec2:CreateTags`, `iam:PassRole`, `iam:GetInstanceProfile`, `pcs:CreateComputeNodeGroup`	
ListComputerNodeGroups	`pcs:ListComputeNodeGroups`	`pcs:GetCluster`
GetComputeNodeGroup	`pcs:GetComputeNodeGroup`	`ec2:DescribeSubnets`
UpdateComputeNodeGroup	`ec2:DescribeVpcs`, `ec2:DescribeSubnets`, `ec2:DescribeSecurityGroups`, `ec2:DescribeLaunchTemplates`, `ec2:DescribeLaunchTemplateVersions`, `ec2:DescribeInstanceTypes`, `ec2:RunInstances`, `ec2:CreateFleet`, `ec2:CreateTags`, `iam:PassRole`, `iam:GetInstanceProfile`, `pcs:UpdateComputeNodeGroup`	`pcs:GetComputeNodeGroup`, `iam:ListInstanceProfiles`, `ec2:DescribeImages`, `pcs:GetCluster`

APIação	Permissões mínimas	Permissões adicionais para o console
DeleteComputeNodeGroup	`pcs:DeleteComputeNodeGroup`	
CreateQueue	`pcs:CreateQueue`	`pcs:ListComputeNodeGroups, pcs:GetCluster`
ListQueues	`pcs:ListQueues`	`pcs:GetCluster`
GetQueue	`pcs:GetQueue`	
UpdateQueue	`pcs:UpdateQueue`	`pcs:ListComputeNodeGroups, pcs:GetQueue`
DeleteQueue	`pcs:DeleteQueue`	

Permissões mínimas necessárias para usar tags

As permissões a seguir são necessárias para usar tags com seus recursos em AWS PCS.

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

Permissões mínimas necessárias para suportar registros

AWS PCSenvia 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 IAM política a seguir especifica as permissões mínimas necessárias para que uma IAM identidade (usuário, grupo ou função) configure e gerencie o AWS PCS serviço.

> **Note**
>
> Os usuários que não configuram e gerenciam o serviço não precisam dessas permissões. Os usuários que executam apenas trabalhos usam secure shell (SSH) para se conectar ao cluster. AWS Identity and Access Management (IAM) não lida com autenticação ou autorização paraSSH.

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

```
            "Resource": "*"
        }
    ]
}
```

Você pode excluir as seguintes permissões da política e, em vez disso, usar a política gerenciada correspondente emIAM:

- `"firehose:*"`

  ```
  AmazonKinesisFirehoseFullAccess
  ```

- `"kms:*"`

  ```
  AWSKeyManagementServicePowerUser
  ```

- `"logs:*"`

  ```
  CloudWatchLogsFullAccess
  ```

- `"s3:*"`

  ```
  AmazonS3FullAccess
  ```

IAMperfis de instância para o AWS Parallel Computing Service

Os aplicativos executados em uma EC2 instância devem incluir AWS credenciais em todas as AWS API solicitações feitas. Recomendamos que você use uma IAM função 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 IAMfunções da Amazon EC2 no Guia do usuário do Amazon Elastic Compute Cloud.

> **Note**
>
> Quando você usa o AWS Management Console para criar uma IAM função para a AmazonEC2, o console cria um perfil de instância automaticamente e dá a ele o mesmo nome da IAM função. Se você usar o AWS CLI, AWS API actions ou an AWS SDK para criar a IAM função, 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 perfil ARN de uma 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

Nome do perfil da instância

O perfil da IAM instância ARN deve começar com `AWSPCS` ou conter `/aws-pcs/` em seu caminho.

Example

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

Permissões

No mínimo, o perfil da instância AWS PCS deve incluir a política a seguir. Ele permite que os nós de computação notifiquem o AWS PCS serviço 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 mazonSSMManaged 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 IAM políticas que suportem seu caso de uso específico.

Criar um perfil da instância

Você pode criar 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.

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 AWS PCS e. IAM

Tópicos

- Não estou autorizado a realizar uma ação em AWS PCS
- Não estou autorizado a realizar iam: PassRole
- Quero permitir que pessoas fora da minha Conta da AWS acessem meus AWS PCS recursos

Não estou autorizado a realizar uma ação em AWS PCS

Se você receber uma mensagem de erro informando que não tem autorização para executar uma ação, é preciso atualizar suas políticas para permitir que você realize a ação.

O exemplo de erro a seguir ocorre quando o `mateojackson` IAM usuário tenta usar o console para ver detalhes sobre um *`my-example-widget`* recurso fictício, mas não tem as permissões fictíciaspcs:*`GetWidget`*.

```
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 transfira uma função para AWS PCS o.

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 IAM usuário 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 Conta da AWS acessem meus AWS PCS recursos

Você pode criar um perfil que os usuários de outras contas ou pessoas fora da sua organização podem usar para acessar seus recursos. Você pode 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 é AWS PCS compatível com esses recursos, consulte[Como o serviço de computação AWS paralela funciona com IAM](#).

- Para saber como fornecer acesso aos seus recursos em todos os Contas da AWS que você possui, consulte Fornecer acesso a um IAM usuário em outro Conta da AWS de sua propriedade no Guia do IAM usuário.
- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte Fornecer Contas da AWS acesso a terceiros no Guia do IAM usuário.
- Para saber como fornecer acesso por meio da federação de identidades, consulte Fornecendo acesso a usuários autenticados externamente (federação de identidades) no Guia do IAMusuário.
- Para saber a diferença entre usar funções e políticas baseadas em recursos para acesso entre contas, consulte Acesso a recursos entre contas IAM no Guia do IAM usuário.

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:

- Guias de início rápido sobre segurança e conformidade — Esses guias de implantação discutem considerações arquitetônicas e fornecem etapas para a implantação de ambientes básicos AWS focados em segurança e conformidade.
- Arquitetura para HIPAA segurança e conformidade na Amazon Web Services — Este whitepaper descreve como as empresas podem usar AWS para criar HIPAA aplicativos qualificados.

> **Note**
>
> Nem todos Serviços da AWS são HIPAA elegíveis. Para obter mais informações, consulte a Referência de serviços HIPAA elegíveis.

- 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, por exemplo PCIDSS, 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 aplicativos e bancos de dados que automaticamente executam o failover entre as zonas sem interrupção. As zonas de disponibilidade são mais 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 API chamadas AWS publicadas para acessar AWS PCS pela rede. Os clientes devem oferecer suporte para:

- Segurança da camada de transporte (TLS). Exigimos TLS 1,2 e recomendamos TLS 1,3.
- Suítes de criptografia com sigilo direto perfeito (), como (Ephemeral PFS Diffie-Hellman) ou DHE (Elliptic Curve Ephemeral 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 uma ID de chave de acesso e uma chave de acesso secreta associada a um IAM principal. Ou você pode usar o AWS Security Token Service (AWS STS) para gerar credenciais de segurança temporárias para assinar solicitações.

Quando 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, AWS PCS crie uma interface de rede elástica (ENI) entre contas em seu. 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 facilitando as operações eficientes HPC de IA/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.

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 em instâncias de grupos de nós.
- Atualize o agendador para mantê-lo dentro das versões compatíveis.
- Autentique e criptografe a comunicação entre os clientes do usuário e os nós aos quais eles se conectam.

Prevenção contra o ataque do “substituto confuso” em todos os serviços

“Confused deputy” é um problema de segurança no qual uma entidade sem permissão para executar uma ação pode coagir uma 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 chaves de contexto de condição `aws:SourceAccount`global `aws:SourceArn`e as chaves de contexto nas políticas de recursos para limitar as permissões que o AWS Parallel Computing Service (AWS PCS) concede a outro serviço ao recurso. Use `aws:SourceArn` se quiser apenas um recurso associado a acessibilidade de serviço. 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 confuso problema do deputado é usar a chave de contexto ARN de condição `aws:SourceArn` global com todo o recurso. Se você não souber a totalidade ARN do recurso ou se estiver especificando vários recursos, use a chave de condição de contexto `aws:SourceArn` global com caracteres curinga (*) para as partes desconhecidas do. ARN Por exemplo, `arn:aws:`*`servicename`*`:*:`*`123456789012`*`:*`.

Se o `aws:SourceArn` valor não contiver o ID da conta, como um bucket do Amazon S3ARN, você deverá usar as duas chaves de contexto de condição global para limitar as permissões.

O valor de `aws:SourceArn` deve ser um clusterARN.

O exemplo a seguir mostra como você pode usar as chaves de contexto de condição `aws:SourceAccount` global `aws:SourceArn` e global 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"
      }
    }
  }
}
```

IAMfunção para EC2 instâncias da Amazon provisionadas como parte de um grupo de nós de computação

AWS PCSorquestra 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 IAM instância por meio do `iamInstanceProfileArn` campo. O perfil da instância especifica as permissões associadas às instâncias provisionadasEC2. AWS PCSaceita 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 IAM identidade (usuário ou função) que cria ou atualiza um grupo de nós de computação. Quando um usuário

chama as `UpdateComputeNodeGroup` API ações `CreateComputeNodeGroup` ou, AWS PCS verifica se o usuário tem permissão para realizar a `iam:PassRole` ação.

O exemplo de política a seguir concede permissões para transmitir somente IAM funções 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.

AMIsegurança relacionada

- Não use AWS PCS amostras 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 das AWS PCS instâncias para reduzir as vulnerabilidades.
- Use AWS Systems Manager para automatizar a aplicação de patches e manter a conformidade com suas políticas de segurança.

- Use somente AWS PCS pacotes oficiais autenticados baixados de AWS fontes oficiais.
- Atualize AWS PCS pacotes regularmente nos nós de computação para receber patches e melhorias de segurança. Considere automatizar esse processo para minimizar as vulnerabilidades.

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 RSS feed de AWS PCS documentação e receber notificações quando uma versão do Slurm se aproximar. EOSL

Monitorar e registrar

- 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 AWS PCS clusters em um local separado VPC para isolar seu HPC ambiente 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 AWS PCS instâncias e sub-redes.
- Use AWS PrivateLink nossos VPC endpoints para manter o tráfego de rede entre seus clusters e outros AWS serviços dentro da AWS rede.

Registro e monitoramento para AWS PCS

O monitoramento é uma parte importante da manutenção da confiabilidade, disponibilidade e desempenho de AWS PCS seus outros AWS recursos. AWSfornece as seguintes ferramentas de monitoramento para observar 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. É possível 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 CPU uso 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 API chamadas 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.

AWS PCSregistros do agendador

Você pode configurar 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. Você pode configurar os registros AWS PCS do agendador usando o AWS PCS console, bem como programaticamente usando o ou. AWS CLI SDK

Sumário

Pré-requisitos

O IAM principal usado para gerenciar o AWS PCS cluster deve permitir`pcs:AllowVendedLogDeliveryForResource`. Aqui está um exemplo AWS IAM de política que permite isso.

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

Configurando registros do agendador usando o console AWS PCS

Para configurar os registros AWS PCS do agendador no console, siga estas etapas:

1. Abra o AWS PCSconsole.
2. Escolha Clusters e navegue até a página de detalhes do AWS PCS cluster em que você habilitará o registro.
3. Escolha Logs.
4. Em entregas de registros — Registros do agendador — opcional

a. Adicione até três destinos de entrega de registros. As opções incluem CloudWatch Logs, Amazon S3 ou Firehose.

b. Escolha Atualizar entregas de registros.

Você pode reconfigurar, adicionar ou remover entregas de registros revisitando esta página.

Configurando os registros do agendador usando o AWS CLI

Para fazer isso, você precisa de pelo menos um destino de entrega, uma fonte de entrega (o PCS cluster) e uma entrega, que é um relacionamento que conecta uma origem a um destino.

Crie um destino de entrega

Você precisa de pelo menos um destino de entrega para receber os registros do agendador de um AWS PCS cluster. Você pode aprender mais sobre esse tópico na PutDeliveryDestination seção do Guia do CloudWatch API usuário.

Para criar um destino de entrega usando o AWS CLI

- Crie um destino com o comando a seguir. Antes da execução do comando, realize as seguintes substituições:
  - Substituir *region-code* com o Região da AWS local onde você criará seu destino. Geralmente, essa será a mesma região em que o AWS PCS cluster está implantado.
  - Substituir *pcs-logs-destination* com seu nome preferido. Ele deve ser exclusivo para todos os destinos de entrega em sua conta.
  - Substituir *resource-arn* com o ARN para um grupo de registros existente no CloudWatch Logs, um bucket do S3 ou um stream de entrega no Firehose. Os exemplos incluem:
    - CloudWatch Grupo de registros

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

    - S3 bucket

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

    - Stream de entrega do Firehose

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

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

Anote o ARN para o novo destino de entrega, pois você precisará dele para configurar as entregas.

Habilite o AWS PCS cluster como fonte de entrega

Para coletar os registros do agendador AWSPCS, configure o cluster como uma fonte de entrega. Para obter mais informações, consulte PutDeliverySourcea Amazon CloudWatch Logs API Reference.

Para configurar um cluster como fonte de entrega usando o AWS CLI

- Ative a entrega de registros do seu cluster com o comando a seguir. Antes da execução do comando, realize as seguintes substituições:
  - Substituir *region-code* com o Região da AWS local onde seu cluster está implantado.
  - Substituir *cluster-logs-source-name* com um nome para essa fonte. Ele deve ser exclusivo para todas as fontes de entrega em seu Conta da AWS. Considere incorporar o nome ou o ID do AWS PCS cluster.
  - Substituir *cluster-arn* com o ARN para o seu AWS PCS cluster

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

Conecte a fonte de entrega do cluster ao destino da entrega

Para que os dados de log do agendador fluam do cluster para o destino, você deve configurar uma entrega que os conecte. Para obter mais informações, consulte CreateDeliverya Amazon CloudWatch Logs API Reference.

Para criar uma entrega usando o AWS CLI

- Crie uma entrega usando o comando a seguir. Antes da execução do comando, realize as seguintes substituições:

- Substituir *region-code* com o Região da AWS local onde sua origem e destino existem.
- Substituir *cluster-logs-source-name* com o nome da sua fonte de entrega acima.
- Substituir *destination-arn* com o ARN de um destino de entrega onde você deseja que os registros sejam entregues.

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

Caminhos e nomes do fluxo de registros do agendador

O caminho e o nome dos registros AWS PCS do agendador dependem do tipo de destino.

- CloudWatch Registros
  - 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 AWS PCS do agendador

AWSPCSos registros do agendador 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": "23.11",
    "scheduler_patch_version": "8",
    "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 pela 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 o AWS PCS uso de monitoramento CloudWatch.

Tópicos

- Monitorando AWS PCS métricas usando CloudWatch
- Monitoramento de AWS PCS instâncias usando a Amazon CloudWatch

Monitorando AWS PCS métricas usando CloudWatch

Você pode monitorar a integridade AWS PCS do cluster 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 PCSpublica as seguintes métricas no PCS namespace AWS/em. CloudWatch Eles têm uma única dimensão,`ClusterId`.

Nome	Descrição	Unidades
ActualCapacity	IdleCapacity + UtilizedCapacity	Contagem
CapacityUtilization	UtilizedCapacity / ActualCapacity	Contagem
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 AWS PCS instâncias usando a Amazon CloudWatch

AWSPCSlança EC2 instâncias da Amazon conforme necessário para atender aos requisitos de escalabilidade definidos em seus grupos de nós de PCS computação. 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 EBS volumes que sustentam a instância. Para reter métricas ou dados de registro das instâncias iniciadas 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 AWS PCS instâncias

Para monitorar instâncias lançadas peloPCS, 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 gerenciadasPCS, consulteEncontrando instâncias de grupos de nós de computação em 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 PCS instâncias.

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 JSON configuração 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 o AWS CLI instalado, 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 PCS específicas 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 autenticaçãoSSH, como 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 PCS computação. 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 AWS CLI comandos a seguir. Antes de executar o comando, faça estas substituições:

- Substituir *DOC-EXAMPLE-BUCKET* com seu próprio nome de bucket S3

Primeiro, (isso é opcional se você tiver um bucket existente), crie um bucket para armazenar seus arquivos de configuração.

```
aws s3 mb s3://DOC-EXAMPLE-BUCKET
```

Em seguida, faça o upload do arquivo para o bucket.

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

Armazenar como SSM parâmetro

Para armazenar seu arquivo como um SSM parâmetro, use o comando a seguir. Antes de executar o comando, faça estas substituições:

- Substituir *region-code* com a AWS região com a qual você está trabalhando AWSPCS.
- (Opcional) Substituir *AmazonCloudWatch-PCS* com seu próprio nome para o parâmetro. Observe que, se você alterar o prefixo do nome de, `AmazonCloudWatch-` precisará adicionar especificamente o acesso de leitura ao SSM parâmetro 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 SSM no.

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:

- *DOC-EXAMPLE-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://DOC-EXAMPLE-BUCKET/config.json /etc/s3-cw-config.json
- /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m
 ec2 -s -c file://etc/s3-cw-config.json

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

O perfil da IAM instância do grupo de nós deve ter acesso ao bucket. Aqui está um exemplo IAM de política 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:::DOC-EXAMPLE-BUCKET",
                "arn:aws:s3:::DOC-EXAMPLE-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 em SSM

Esse script instala o CloudWatch agente, importa um arquivo de configuração de um SSM parâmetro e inicia o CloudWatch agente com ele. Substitua os seguintes valores nesse script pelos seus próprios detalhes:

- (Opcional) Substituir *AmazonCloudWatch-PCS* com seu próprio nome para o parâmetro.

```
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 IAM instância 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 SSM parâmetro no perfil da instância do seu grupo de nós. Aqui está um exemplo de IAM política que ilustra isso para o 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 os CloudWatch endpoints SSM e. Isso pode ser feito usando grupos de segurança ou VPC endpoints, dependendo da arquitetura do cluster.

Registrando API chamadas do serviço de computação AWS paralela usando AWS CloudTrail

AWS PCSé integrado com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço em AWS PCS. CloudTrail captura todas as API chamadas para AWS PCS eventos. As chamadas capturadas incluem chamadas do AWS PCS console e chamadas de código para as AWS PCS API operações. 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 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 PCSinformações em CloudTrail

CloudTrail é ativado no seu Conta da AWS quando você cria a conta. Quando a atividade ocorre em 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 sua 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 as informações a seguir.

- Visão geral da criação de uma trilha
- CloudTrail serviços e integrações suportados
- Configurando SNS notificações da Amazon para CloudTrail
- Recebendo arquivos de CloudTrail log de várias regiões e Recebendo arquivos de CloudTrail log de várias contas

Todas as AWS PCS ações são registradas CloudTrail e documentadas na APIReferência 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:

- 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 um perfil ou de um usuário federado.
- Se a solicitação foi feita por outro AWS serviço.

Para obter mais informações, consulte o CloudTrail userIdentityelemento.

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 API chamadas públicas, 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 AWS PCS

As seções a seguir descrevem os endpoints e as cotas de serviço do AWS Parallel Computing Service ()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 . É possível solicitar aumentos para algumas cotas e outras cotas 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 (Norte da Virgínia)	us-east-1	pcs.us-east-1.amazonaws.com	HTTPS
Leste dos EUA (Ohio)	us-east-2	pcs.us-east-2.amazonaws.com	HTTPS
Oeste dos EUA (Oregon)	us-west-2	pcs.us-west-2.amazonaws.com	HTTPS
Ásia-Pacífico (Singapura)	ap-southeast-1	pcs.ap-southeast-1.amazonaws.com	HTTPS
Ásia-Pacífico (Sydney)	ap-southeast-2	pcs.ap-southeast-2.amazonaws.com	HTTPS

Nome da região	Região	Endpoint	Protocolo
Ásia-Pacífico (Tóquio)	ap-northeast-1	pcs.ap-northeast-1.amazonaws.com	HTTPS
Europa (Frankfurt)	eu-central-1	pcs.eu-central-1.amazonaws.com	HTTPS
Europa (Irlanda)	eu-west-1	pcs.eu-west-1.amazonaws.com	HTTPS
Europa (Estocolmo)	eu-north-1	pcs.eu-north-1.amazonaws.com	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.

Cotas relevantes para outros serviços AWS

AWS PCSusa outros AWS serviços. Suas cotas de serviço para esses serviços afetam seu uso de AWS PCS.

Cotas EC2 de serviços da Amazon que impactam AWS PCS

- Solicitações de instância spot
- Executando instâncias sob demanda
- Modelos de inicialização
- Versões do modelo de execução
- EC2APISolicitações da Amazon

Para obter mais informações, consulte as cotas EC2 de serviços da Amazon no Guia do usuário do Amazon Elastic Compute Cloud.

Notas de lançamento para AWS PCS amostra AMIs

AWS PCSAMIsAs amostras têm uma cadência de lançamento noturno de patches de segurança. Esses patches de segurança incrementais não estão incluídos nas notas de versão oficiais.

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

Sumário

AWS PCSamostra x86_64 AMI para Slurm 23.11 (Amazon Linux 2)

Este documento descreve as últimas alterações, adições, problemas conhecidos e correções do AWS PCS Sample x86_64 AMI (Amazon Linux 2).

- Data de criação: 15 de julho de 2024
- Data de lançamento: 22 de agosto de 2024
- Última atualização: 22 de agosto de 2024

AMInome

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

EC2Instâncias suportadas

- Todas as instâncias com um processador x86 de 64 bits. Para encontrar instâncias compatíveis, navegue até o EC2console da Amazon. Escolha Tipos de instância e, em seguida, pesquise por`Architectures=x86_64`.

AMIconteúdos

- AWS Serviço suportado: AWS PCS
- Sistema operacional: Amazon Linux 2
- Arquitetura de computação: x86_64
- Kernel Linux: 5.10.220-209.867.amzn2.x86_64
- EBStipo de volume: gp2
- AWS PCSInstalador do Slurm 23.11:23.11.9-1
- AWS PCSinstalador de software: 1.0.0-1
- EFAInstalador: 1.33.0
- GDRCopy: 2,4
- NVIDIADriver: 535.154.05
- NVIDIACUDA: 12.2.2_535.104.05

Avisos

- Nenhum

Data de lançamento: 2024-08-22

Atualizado

- Nenhum. Primeiro lançamento.

Adicionado

- Nenhum. Primeiro lançamento.

Removido

- Nenhum. Primeiro lançamento.

AWS PCSamostra de Arm64 AMI para Slurm 23.11 (Amazon Linux 2)

Este documento descreve as últimas alterações, adições, problemas conhecidos e correções do AWS PCS Sample Arm64 (AMIAmazon Linux 2).

- Data de criação: 15 de julho de 2024
- Data de lançamento: 22 de agosto de 2024
- Última atualização: 22 de agosto de 2024

AMInome

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

EC2Instâncias suportadas

- Todas as instâncias com um processador Arm de 64 bits. Para encontrar instâncias compatíveis, navegue até o EC2console da Amazon. Escolha Tipos de instância e, em seguida, pesquise por`Architectures=arm64`.

AMIconteúdos

- AWS Serviço suportado: AWS PCS
- Sistema operacional: Amazon Linux 2
- Arquitetura de computação: arm64
- Kernel Linux: 5.10.220-209.867.amzn2.aarch64
- EBStipo de volume: gp2
- AWS PCSInstalador do Slurm 23.11:23.11.9-1
- AWS PCSinstalador de software: 1.0.0-1
- EFAInstalador: 1.33.0
- GDRCopy: 2,4
- NVIDIADriver: 535.154.05
- NVIDIACUDA: 12.2.2_535.104.05

Avisos

- Nenhum

Data de lançamento: 2024-08-22

Atualizado

- Nenhum. Primeiro lançamento.

Adicionado

- Nenhum. Primeiro lançamento.

Removido

- Nenhum. Primeiro lançamento.

Histórico do documento para o Guia do usuário do AWS PCS

A tabela a seguir descreve as versões de documentação do AWS PCS.

Data	Alteração	Atualizações feitas na documentação	APIversões 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 PCSsoltar	Versão inicial do guia AWS PCS do usuário.	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.