Guide de l’utilisateur

AWS PCS

AWS PCS: Guide de l’utilisateur

Table of Contents

Qu'est-ce que le AWS PCS ? 1
Concepts 1
Commencez avec AWS PCS 3
Prérequis 5
Inscrivez-vous AWS et créez un utilisateur administratif 5
Installez le AWS CLI pour AWS PC 7
Autorisations IAM requises 7
En utilisant CloudFormation 8
Créer un VPC et des sous-réseaux 8
Trouvez le groupe de sécurité par défaut pour le VPC du cluster 10
Création de groupes de sécurité 10
Créer les groupes de sécurité 10
Créer un cluster 11
Création d'un espace de stockage partagé dans Amazon EFS 12
Créer un espace de stockage partagé dans FSx pour Lustre 13
Création de groupes de nœuds de calcul 15
Création d'un profil d'instance 15
Créer des modèles de lancement 17
Création d'un groupe de nœuds de calcul pour les nœuds de connexion 18
Création d'un groupe de nœuds de calcul pour les tâches 19
Créer une file d'attente 20
Connectez-vous à votre cluster 21
Explorez l'environnement du cluster 23
Changer d'utilisateur 23
Travailler avec des systèmes de fichiers partagés 23
Interagir avec Slurm 24
Exécuter une tâche sur un seul nœud 24
Exécuter une tâche MPI multi-nœuds avec Slurm 26
Supprimer vos AWS ressources 29
Commencez avec CloudFormation AWS PCS 32
CloudFormation À utiliser pour créer un cluster 32
Connexion à un cluster 34
Nettoyer un cluster 35
Éléments d'un CloudFormation modèle pour AWS PCS 35

En-tête ... 36
Métadonnées ... 36
Parameters ... 37
Mappages ... 39
Ressources ... 39
Sorties ... 43
Modèles pour créer un cluster d'échantillons ... 44
Clusters ... 46
Création d'un cluster ... 46
Conditions préalables ... 47
Création d'un cluster AWS PCS ... 47
Mise à jour d'un cluster ... 52
Avantages des mises à jour du cluster ... 52
Modifications de configuration prises en charge ... 52
Limites ... 52
Conditions préalables pour les mises à jour du cluster ... 53
Processus de mise à jour et impact sur le travail ... 53
Facturation lors des mises à jour ... 53
Mise à jour d'un cluster ... 54
FAQ ... 56
Dépannage ... 57
Suppression d'un cluster ... 58
Considérations relatives à la suppression d'un cluster AWS PCS ... 58
Supprimer le cluster ... 58
Taille du cluster ... 59
Secrets du cluster ... 60
AWS Secrets Manager À utiliser pour trouver le secret du cluster ... 61
Utilisez AWS PCS pour trouver le secret du cluster ... 62
Obtenez le secret du cluster Slurm ... 63
Rotation secrète ... 64
Groupes de nœuds de calcul ... 69
Création d'un groupe de nœuds de calcul ... 69
Conditions préalables ... 70
Création d'un groupe de nœuds de calcul dans AWS PCS ... 70
Mise à jour d'un groupe de nœuds de calcul ... 76
Options de mise à jour d'un groupe de nœuds de calcul AWS PCS ... 76

Considérations relatives à la mise à jour d'un groupe de nœuds de calcul AWS PCS 77
Pour mettre à jour un groupe de nœuds de calcul AWS PCS 78
Suppression d'un groupe de nœuds de calcul 80
Considérations relatives à la suppression d'un groupe de nœuds de calcul 80
Supprimer le groupe de nœuds de calcul 81
Obtenir des informations sur les groupes de nœuds de calcul 82
Recherche d'instances de groupes de nœuds de calcul 85
Utilisation de modèles de lancement 88
Présentation de 88
Créer un modèle de lancement de base 90
Utilisation des données utilisateur Amazon EC2 92
Exemple : installation d'un logiciel à partir d'un référentiel de packages 94
Exemple : exécution de scripts à partir d'un compartiment S3 95
Exemple : définir des variables d'environnement globales 96
Exemple : utilisation d'un système de fichiers EFS comme répertoire de base partagé 97
Réserve de capacité 98
Utilisation ODCRs avec AWS PCS 99
Blocs de capacité 101
Paramètres utiles du modèle de lancement 107
Activez la CloudWatch surveillance détaillée 107
Service de métadonnées d'instance, version 2 (IMDS v2) 108
Files d'attente 110
Création d'une file d'attente 110
Prérequis 110
Pour créer une file d'attente dans AWS PCS 111
Mettre à jour une file d'attente 113
Considérations relatives à la mise à jour d'une file AWS PCS 113
Pour mettre à jour une file d'attente AWS PCS 113
Suppression d'une file d'attente 115
Considérations relatives à la suppression d'une file d'attente 116
Supprimer la file d'attente 116
Nœuds de connexion 118
Utilisation d'un groupe de nœuds de calcul pour la connexion 118
Création d'un groupe de nœuds de calcul AWS PCS pour les nœuds de connexion 118
Mise à jour d'un groupe de nœuds de calcul AWS PCS pour les nœuds de connexion 119
Suppression d'un groupe de nœuds de calcul AWS PCS pour les nœuds de connexion 120

Utilisation d'instances autonomes comme nœuds de connexion 120
Étape 1 — Récupérez l'adresse et le secret du cluster AWS PCS cible 121
Étape 2 — Lancer une instance EC2 122
Étape 3 — Installation de Slurm sur l'instance 123
Étape 4 — Récupérez et stockez le secret du cluster 123
Étape 5 — Configuration de la connexion au cluster AWS PCS 124
Étape 6 — (Facultatif) Testez la connexion 126
Connexion d'un nœud de connexion autonome à plusieurs clusters 127
Conditions préalables 128
Code du script 129
Utilisation du script 137
Réseaux 140
Exigences requises pour le VPC et les sous-réseaux 140
Exigences et considérations requises pour le VPC 140
Exigences et considérations requises pour les sous-réseaux 142
Création d'un VPC 143
Conditions préalables 144
Création d'un Amazon VPC 144
Groupes de sécurité 146
Exigences de groupe de sécurité 146
Plusieurs interfaces réseau 148
Groupes de placement 149
Utilisation d'Elastic Fabric Adapter (EFA) 150
Identifier les instances EC2 compatibles avec EFA 151
Création d'un groupe de sécurité pour prendre en charge les communications EFA 152
(Facultatif) Créez un groupe de placement 153
Création ou mise à jour d'un modèle de lancement EC2 153
Création ou mise à jour de groupes de nœuds de calcul pour EFA 154
(Facultatif) Testez EFA 155
(Facultatif) Utilisez un CloudFormation modèle pour créer un modèle de lancement compatible avec l'EFA 157
Systèmes de fichiers réseau 159
Considérations relatives à l'utilisation de systèmes de fichiers réseau 159
Exemples de montages réseau 160
Images de machines Amazon (AMIs) 166
Utilisation d'un échantillon AMIs 166

Trouver un échantillon AWS PCS actuel AMIs 167
En savoir plus sur l'échantillon AWS PCS AMIs 168
Créez le vôtre AMIs compatible avec les AWS PCS 168
Personnalisé AMIs 168
Étape 1 — Lancer une instance temporaire 170
Étape 2 — Installation de l'agent AWS PCS 170
Étape 3 — Installation de Slurm 173
Étape 4 — (Facultatif) Installation de pilotes, de bibliothèques et de logiciels d'application supplémentaires 176
Étape 5 — Création d'une AMI compatible avec AWS PCS 177
Étape 6 — Utiliser l'AMI personnalisée avec un groupe de nœuds de calcul AWS PCS 178
Étape 7 — Mettre fin à l'instance temporaire 180
Installateurs à construire AMIs 180
AWS Programme d'installation du logiciel PCS Agent 180
Installateur Slurm 181
Systèmes d'exploitation pris en charge 182
Types d'instance pris en charge 182
Versions de Slurm prises en charge 182
Vérifiez les installateurs à l'aide d'une somme de contrôle 182
Notes de mise à jour pour AMIs 189
Exemple AMIs pour x86_64 () AL2 189
Exemple AMIs pour Arm64 () AL2 192
Systèmes d'exploitation pris en charge 196
AWS Versions de l'agent PCS 198
Slurm 202
Versions Slurm 202
Versions de Slurm prises en charge sur PCS AWS 203
Versions de Slurm non prises en charge sur PCS AWS 204
Notes de mise à jour 204
Questions fréquentes (FAQ) 207
Comptabilité Slurm 209
Modification des paramètres comptables 210
Concepts clés 210
Obtenez la configuration comptable d'un cluster AWS PCS existant 212
API REST de Slurm 213
Cas d'utilisation courants 213

Exigences et limitations 213
Activer l'API REST 214
Authentification par API REST 216
Utiliser l'API REST 221
FAQ SUR L'API REST 223
Redémarrage de Slurm 226
Avantages du redémarrage de Slurm 226
Quand utiliser le redémarrage de Slurm 226
Limitations 227
Redémarrer un nœud de calcul 227
Annuler le redémarrage 228
FAQ 229
Résolution des problèmes 231
Réglages personnalisés de Slurm 232
Avantages des paramètres personnalisés de Slurm 232
Configuration de paramètres personnalisés 232
Validation et gestion des erreurs 234
Limitations 234
Paramètres du cluster 235
Paramètres du groupe de nœuds de calcul 237
Paramètres de file d'attente 237
Résolution des problèmes 238
Plug-ins SPANK 239
Installez les plugins SPANK 240
Configurer les plugins SPANK 240
FAQ sur les plugins SPANK 242
Plug-ins de filtre CLI Slurm 242
Exigences 243
Limites et considérations de sécurité 243
Configurer les plugins de filtre CLI 243
Utilisation d'Amazon S3 pour déployer un script de plug-in de filtrage CLI 247
Traduire un script du plugin Job Submit 249
FAQ 250
Résolution des problèmes 252
Sécurité 254
Protection des données 255

Chiffrement au repos 256
Chiffrement en transit 256
Gestion des clés 257
Confidentialité du trafic inter-réseaux 257
Chiffrer le trafic des API 258
Chiffrement du trafic de données 258
Politique relative aux clés KMS pour les volumes EBS chiffrés 258
Points de terminaison d'interface VPC ()AWS PrivateLink 265
Considérations 265
Création d'un point de terminaison d'interface 265
Création d'une politique de point de terminaison 266
Gestion de l'identité et des accès 267
Public ciblé 268
Authentification par des identités 268
Gestion de l'accès à l'aide de politiques 270
Comment fonctionne AWS Parallel Computing Service avec IAM 271
Exemples de politiques basées sur l'identité 277
AWS politiques gérées 281
Rôles liés à un service 283
Rôle EC2 Spot 285
Autorisations minimales 286
Profils d'instance 294
Résolution des problèmes 298
Validation de conformité 300
Résilience 301
Sécurité de l'infrastructure 301
Analyse et gestion des vulnérabilités 302
Prévention du problème de l'adjoint confus entre services 303
Rôle IAM pour les instances Amazon EC2 mises en service dans le cadre d'un groupe de nœuds de calcul 304
Bonnes pratiques de sécurité 305
Sécurité liée à l'AMI 305
Sécurité de Slurm Workload Manager 305
Surveillance et journalisation 306
Sécurité du réseau 306
Journalisation et surveillance 307

Journaux d'achèvement des tâches 307
Conditions préalables 308
Configurer les journaux d'achèvement des tâches 309
Comment trouver les journaux d'achèvement des tâches 311
Champs du journal d'achèvement des tâches 311
Exemples de journaux d'achèvement des tâches 315
Journaux du planificateur 318
Conditions préalables 319
Configurer les journaux du planificateur 319
Le planificateur enregistre les chemins et les noms des flux 321
Exemple d'enregistrement du journal du planificateur 322
Surveillance avec CloudWatch 323
Surveillance des métriques 323
Surveillance des instances 324
CloudTrail journaux 333
AWS Informations PCS dans CloudTrail 334
Comprendre les entrées des fichiers CloudTrail journaux à partir de AWS PCS 335
Points de terminaison et quotas de service 337
Points de terminaison de service 337
Quotas de service 340
Quotas internes 341
Quotas pertinents pour les autres AWS services 341
Résolution des problèmes 343
L'instance EC2 est arrêtée et remplacée après le redémarrage 343
Résoudre les problèmes d'amorçage et d'enregistrement des nœuds de calcul dans les PCS AWS 344
Comment fonctionne Slurm sur PC AWS 345
Récupérez les journaux d'instance 346
Récupérer VPC/Subnet/Security des groupes à partir d'un ID d'instance 347
Problèmes d'enregistrement des nœuds 348
Problèmes de jointure au cluster Slurm 351
Historique de la documentation 354
AWS Glossaire 382
..... ccclxxxiii

Qu'est-ce que le service de calcul AWS parallèle ?

AWS Le Parallel Computing Service (AWS PCS) est un service géré qui facilite l'exécution et le dimensionnement des charges de travail de calcul haute performance (HPC), ainsi que la création de modèles scientifiques et d'ingénierie basés sur AWS l'utilisation de Slurm. Utilisez AWS PCS pour créer des clusters de calcul qui intègrent les meilleurs systèmes de AWS calcul, de stockage, de mise en réseau et de visualisation. Exécutez des simulations ou créez des modèles scientifiques et techniques. Rationalisez et simplifiez les opérations de votre cluster à l'aide de fonctionnalités de gestion et d'observabilité intégrées. Donnez à vos utilisateurs les moyens de se concentrer sur la recherche et l'innovation en leur permettant d'exécuter leurs applications et leurs tâches dans un environnement familier.

Rubriques

- Concepts en AWS PCS

Concepts en AWS PCS

Un cluster dans AWS PCS possède une ou plusieurs files d'attente associées à au moins un groupe de nœuds de calcul. Les tâches sont soumises à des files d'attente et exécutées sur des EC2 instances définies par des groupes de nœuds de calcul. Vous pouvez utiliser ces bases pour implémenter des architectures HPC sophistiquées.

Cluster

Un cluster est une ressource permettant de gérer des ressources et d'exécuter des charges de travail. Un cluster est une ressource AWS PCS qui définit un assemblage de configuration de calcul, de mise en réseau, de stockage, d'identité et de planificateur de tâches. Vous créez un cluster en spécifiant le planificateur de tâches que vous souhaitez utiliser (Slurm actuellement), la configuration du planificateur que vous souhaitez, le contrôleur de service que vous souhaitez gérer le cluster et le VPC dans lequel vous souhaitez que les ressources du cluster soient lancées. Le planificateur accepte et planifie les tâches, et lance également les nœuds de calcul (EC2 instances) qui traitent ces tâches.

Groupe de nœuds de calcul

Un groupe de nœuds de calcul est un ensemble de nœuds de calcul que AWS PCS utilise pour exécuter des tâches ou fournir un accès interactif à un cluster. Lorsque vous définissez un groupe

de nœuds de calcul, vous spécifiez des caractéristiques communes telles que les types d' EC2 instances Amazon, le nombre minimal et maximal d'instances, les sous-réseaux VPC cibles, Amazon Machine Image (AMI), les options d'achat et la configuration de lancement personnalisée. AWS PCS utilise ces paramètres pour lancer, gérer et arrêter efficacement les nœuds de calcul d'un groupe de nœuds de calcul.

File d’attente

Lorsque vous souhaitez exécuter une tâche sur un cluster spécifique, vous la soumettez à une file d'attente spécifique (parfois appelée partition). La tâche reste dans la file d'attente jusqu'à ce que AWS PCS la planifie pour qu'elle s'exécute sur un groupe de nœuds de calcul. Vous associez un ou plusieurs groupes de nœuds de calcul à chaque file d'attente. Une file d'attente est requise pour planifier et exécuter des tâches sur les ressources du groupe de nœuds de calcul sous-jacents à l'aide des différentes politiques de planification proposées par le planificateur de tâches. Les utilisateurs ne soumettent pas de tâches directement à un nœud de calcul ou à un groupe de nœuds de calcul.

Administrateur système

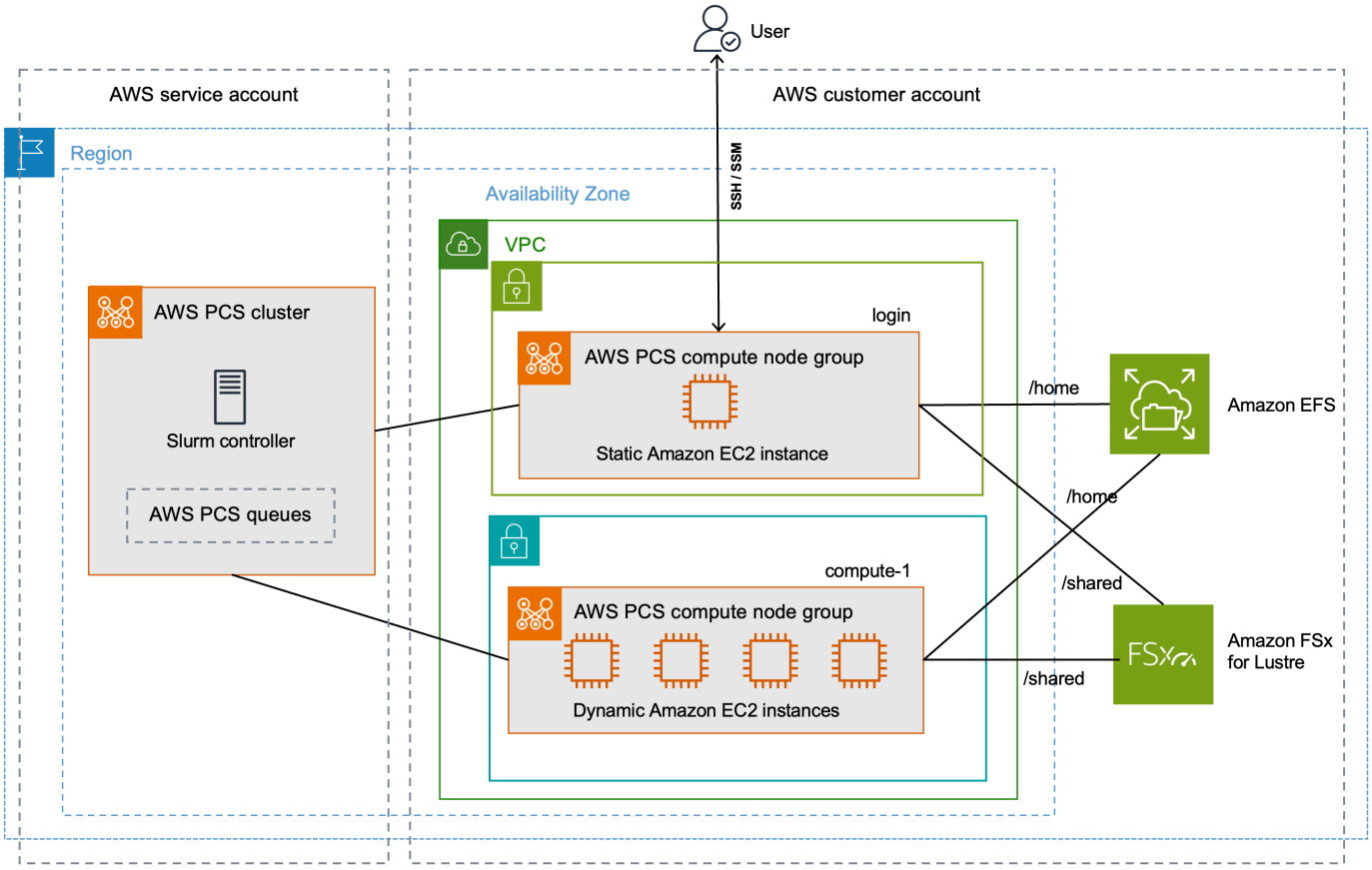
Un administrateur système déploie, gère et exploite un cluster. Ils peuvent accéder à AWS PCS via AWS Management Console l'API AWS PCS et le AWS SDK. Ils ont accès à des clusters spécifiques via SSH ou AWS Systems Manager, où ils peuvent exécuter des tâches administratives, exécuter des tâches, gérer des données et effectuer d'autres activités basées sur le shell. Pour plus d'informations, consultez la documentation AWS Systems Manager.

Utilisateur final

L'utilisateur final n'a pas day-to-day la responsabilité de déployer ou d'exploiter un cluster. Ils utilisent une interface de terminal (telle que SSH) pour accéder aux ressources du cluster, exécuter des tâches, gérer les données et effectuer d'autres activités basées sur le shell.

Commencez avec AWS Parallel Computing Service

Il s'agit d'un didacticiel pour créer un cluster simple que vous pouvez utiliser pour essayer AWS PCS. La figure suivante montre la conception du cluster.



La conception du cluster du didacticiel comporte les éléments clés suivants :

- Un VPC et des sous-réseaux qui répondent aux exigences du réseau AWS PCS.
- Un système de fichiers Amazon EFS, qui sera utilisé comme répertoire de base partagé.
- Un système de fichiers Amazon FSx for Lustre, qui fournit un répertoire partagé à hautes performances.
- Un cluster AWS PCS, qui fournit un contrôleur Slurm.
- 2 groupes de nœuds de calcul AWS PCS.
  - Le groupe de `login` nœuds, qui fournit un accès interactif au système basé sur le shell.
  - Le groupe de `compute-1` nœuds fournit des instances évolutives de manière élastique pour exécuter des tâches.

- 1 file d'attente qui envoie des tâches aux EC2 instances du groupe de `compute-1` nœuds.

Le cluster nécessite des AWS ressources supplémentaires, telles que des groupes de sécurité, des rôles IAM et des modèles de EC2 lancement, qui ne sont pas illustrés dans le schéma.

> **Note**
>
> Nous vous recommandons de suivre les étapes de ligne de commande décrites dans cette rubrique dans un shell Bash. Si vous n'utilisez pas de shell Bash, certaines commandes de script telles que les caractères de continuation de ligne et la façon dont les variables sont définies et utilisées nécessitent un ajustement pour votre shell. En outre, les règles de votre shell en matière de guillemets peuvent être différentes. Pour plus d'informations, voir Guillemets et littéraux avec chaînes AWS CLI dans le Guide de l'AWS Command Line Interface utilisateur de la version 2.

Rubriques

- Conditions préalables pour démarrer avec PCS AWS
- Utilisation AWS CloudFormation avec le didacticiel AWS PCS
- Création d'un VPC et de sous-réseaux pour PCS AWS
- Création de groupes de sécurité pour AWS PCS
- Création d'un cluster dans AWS PCS
- Créez un espace de stockage partagé pour les AWS PC dans Amazon Elastic File System
- Créez un espace de stockage partagé pour les AWS PC dans Amazon FSx for Lustre
- Création de groupes de nœuds de calcul dans AWS PCS
- Créez une file d'attente pour gérer les tâches dans AWS PCS
- Connectez-vous à votre cluster AWS PCS
- Explorez l'environnement de cluster dans AWS PCS
- Exécuter une tâche à nœud unique dans AWS PCS
- Exécuter une tâche MPI multi-nœuds avec Slurm sur PCS AWS
- Supprimer vos AWS ressources pour AWS PCS

Conditions préalables pour démarrer avec PCS AWS

Consultez les rubriques suivantes pour préparer votre environnement de développement Compte AWS et celui de votre environnement de développement local pour AWS PCS.

Rubriques

- Inscrivez-vous AWS et créez un utilisateur administratif
- Installez le AWS CLI pour AWS PC
- Autorisations IAM requises pour PCS AWS

Inscrivez-vous AWS et créez un utilisateur administratif

Effectuez les tâches suivantes pour configurer le service de calcul AWS parallèle (AWS PCS).

Rubriques

- Inscrivez-vous pour un Compte AWS
- Création d'un utilisateur doté d'un accès administratif

Inscrivez-vous pour un Compte AWS

Si vous n'en avez pas Compte AWS, procédez comme suit pour en créer un.

Pour vous inscrire à un Compte AWS

1. Ouvrez l'https://portal.aws.amazon.com/billing/inscription.
2. Suivez les instructions en ligne.

   Dans le cadre de la procédure d'inscription, vous recevrez un appel téléphonique ou un SMS et vous saisirez un code de vérification en utilisant le clavier numérique du téléphone.

   Lorsque vous vous inscrivez à un Compte AWS, un Utilisateur racine d'un compte AWSest créé. Par défaut, seul l'utilisateur racine a accès à l'ensemble des Services AWS et des ressources de ce compte. La meilleure pratique de sécurité consiste à attribuer un accès administratif à un utilisateur, et à utiliser uniquement l'utilisateur racine pour effectuer les tâches nécessitant un accès utilisateur racine.

AWS vous envoie un e-mail de confirmation une fois le processus d'inscription terminé. À tout moment, vous pouvez consulter l'activité actuelle de votre compte et gérer votre compte en accédant à https://aws.amazon.com/et en choisissant Mon compte.

Création d'un utilisateur doté d'un accès administratif

Après vous être inscrit à un Compte AWS, sécurisez Utilisateur racine d'un compte AWS AWS IAM Identity Center, activez et créez un utilisateur administratif afin de ne pas utiliser l'utilisateur root pour les tâches quotidiennes.

Sécurisez votre Utilisateur racine d'un compte AWS

1. Connectez-vous en AWS Management Consoletant que propriétaire du compte en choisissant Utilisateur root et en saisissant votre adresse Compte AWS e-mail. Sur la page suivante, saisissez votre mot de passe.

   Pour obtenir de l'aide pour vous connecter en utilisant l'utilisateur racine, consultez Connexion en tant qu'utilisateur racine dans le Guide de l'utilisateur Connexion à AWS .
2. Activez l'authentification multifactorielle (MFA) pour votre utilisateur racine.

   Pour obtenir des instructions, voir Activer un périphérique MFA virtuel pour votre utilisateur Compte AWS root (console) dans le guide de l'utilisateur IAM.

Création d'un utilisateur doté d'un accès administratif

1. Activez IAM Identity Center.

   Pour obtenir des instructions, consultez Activation d' AWS IAM Identity Center dans le Guide de l'utilisateur AWS IAM Identity Center .
2. Dans IAM Identity Center, octroyez un accès administratif à un utilisateur.

   Pour un didacticiel sur l'utilisation du Répertoire IAM Identity Center comme source d'identité, voir Configurer l'accès utilisateur par défaut Répertoire IAM Identity Center dans le Guide de AWS IAM Identity Center l'utilisateur.

Connexion en tant qu'utilisateur doté d'un accès administratif

- Pour vous connecter avec votre utilisateur IAM Identity Center, utilisez l'URL de connexion qui a été envoyée à votre adresse e-mail lorsque vous avez créé l'utilisateur IAM Identity Center.

Pour obtenir de l'aide pour vous connecter en utilisant un utilisateur d'IAM Identity Center, consultez la section Connexion au portail AWS d'accès dans le guide de l'Connexion à AWS utilisateur.

Attribution d'un accès à d'autres utilisateurs

1. Dans IAM Identity Center, créez un ensemble d'autorisations qui respecte la bonne pratique consistant à appliquer les autorisations de moindre privilège.

   Pour obtenir des instructions, consultez Création d'un ensemble d'autorisations dans le Guide de l'utilisateur AWS IAM Identity Center .
2. Attribuez des utilisateurs à un groupe, puis attribuez un accès par authentification unique au groupe.

   Pour obtenir des instructions, consultez Ajout de groupes dans le Guide de l'utilisateur AWS IAM Identity Center .

Installez le AWS CLI pour AWS PC

Vous devez utiliser la dernière version du AWS CLI. Pour plus d'informations, voir Installation ou mise à jour vers la dernière version du AWS CLI dans le guide de AWS Command Line Interface l'utilisateur de la version 2.

Vous devez configurer le AWS CLI. Pour plus d'informations, voir Configurer le AWS CLI dans le guide de AWS Command Line Interface l'utilisateur de la version 2.

Entrez la commande suivante à une invite de commande pour vérifier votre AWS CLI ; elle devrait afficher des informations d'aide.

```
aws pcs help
```

Autorisations IAM requises pour PCS AWS

Le principal de sécurité IAM que vous utilisez doit être autorisé à utiliser les rôles AWS PCS IAM, les rôles liés à un service AWS CloudFormation, un VPC et les ressources associées. Pour plus d'informationsIdentity and Access Management pour le service de calcul AWS parallèle, voir et Créer un rôle lié à un service dans le Guide de l'Gestion des identités et des accès AWS utilisateur. Vous

devez effectuer toutes les étapes de ce guide avec le même utilisateur. Exécutez la commande suivante pour vérifier l'utilisateur actuel :

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

Utilisation AWS CloudFormation avec le didacticiel AWS PCS

Le didacticiel AWS PCS comporte de nombreuses étapes et vise à vous aider à comprendre les composants d'un cluster AWS PCS et les procédures requises pour le créer. Nous vous recommandons de suivre les étapes du didacticiel au moins une fois. Une fois que vous aurez bien compris ce que cela implique, vous pouvez l'utiliser AWS CloudFormation pour créer rapidement le cluster d'échantillons grâce à l'automatisation.

CloudFormation est un AWS service qui vous permet de créer et de provisionner des déploiements AWS d'infrastructure de manière prévisible et répétée. Vous pouvez utiliser un CloudFormation modèle pour provisionner automatiquement les AWS ressources du cluster d'échantillons sous la forme d'une unité unique, appelée pile. Vous pouvez supprimer la pile lorsque vous en avez terminé.

Pour de plus amples informations, veuillez consulter Commencez avec CloudFormation AWS PCS.

Création d'un VPC et de sous-réseaux pour PCS AWS

Vous pouvez créer un VPC et des sous-réseaux à l'aide d'un modèle. CloudFormation Utilisez l'URL suivante pour télécharger le CloudFormation modèle, puis chargez-le dans la CloudFormation console pour créer une nouvelle CloudFormation pile. Pour plus d'informations, consultez la section Utilisation de la CloudFormation console dans le guide de AWS CloudFormation l'utilisateur.

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

Le modèle étant ouvert dans la CloudFormation console, entrez les options suivantes. Vous pouvez utiliser les valeurs par défaut fournies dans le modèle.

- Sous Fournir un nom de pile :
  - Sous Nom de la pile, entrez :

    ```
    hpc-networking
    ```

- Sous Paramètres :
  - Dans le cadre du VPC :
    - Sous CidrBlock, entrez :

```
10.3.0.0/16
```

  - Sous les sous-réseaux A :
    - Sous CidrPublicSubnetA, entrez :

```
10.3.0.0/20
```

    - Sous CidrPrivateSubnetA, entrez :

```
10.3.128.0/20
```

  - Sous les sous-réseaux B :
    - Sous CidrPublicSubnetB, entrez :

```
10.3.16.0/20
```

    - Sous CidrPrivateSubnetB, entrez :

```
10.3.144.0/20
```

  - Sous les sous-réseaux C :
    - Pour ProvisionSubnetsC, sélectionnez Vrai
    - Sous CidrPublicSubnetC, entrez :

```
10.3.32.0/20
```

    - Sous CidrPrivateSubnetC, entrez :

```
10.3.160.0/20
```

- Sous Capacités :
  - Cochez la case Je reconnais que cela AWS CloudFormation pourrait créer des ressources IAM.

Surveillez l'état de la CloudFormation pile. Lorsqu'il atteint`CREATE_COMPLETE`, trouvez l'ID du groupe de sécurité par défaut dans le nouveau VPC. Vous utiliserez l'identifiant ultérieurement dans le didacticiel.

Trouvez le groupe de sécurité par défaut pour le VPC du cluster

Pour trouver l'ID du groupe de sécurité par défaut dans le nouveau VPC, suivez cette procédure :

- Accédez à la console Amazon VPC.
- Dans le tableau de bord VPC, sélectionnez Filtrer par VPC.
  - Choisissez le VPC dont le nom commence par. `hpc-networking`
  - Sous Sécurité, sélectionnez Groupes de sécurité.
- Trouvez l'ID du groupe de sécurité pour le groupe nommé`default`. Il contient la description`default VPC security group`. Vous utiliserez l'ID ultérieurement pour configurer les modèles de lancement EC2.

Création de groupes de sécurité pour AWS PCS

AWS Le PCS s'appuie sur des groupes de sécurité pour gérer le trafic réseau entrant et sortant d'un cluster et de ses groupes de nœuds de calcul. Pour des informations détaillées sur ce sujet, voirExigences et considérations relatives aux groupes de sécurité.

Au cours de cette étape, vous allez utiliser un CloudFormation modèle pour créer deux groupes de sécurité.

- Un groupe de sécurité du cluster, qui permet les communications entre le contrôleur AWS PCS, les nœuds de calcul et les nœuds de connexion.
- Un groupe de sécurité SSH entrant, que vous pouvez éventuellement ajouter à vos nœuds de connexion pour prendre en charge l'accès SSH

Création des groupes de sécurité pour AWS PCS

Vous pouvez utiliser un CloudFormation modèle pour créer les groupes de sécurité. Utilisez l'URL suivante pour télécharger le CloudFormation modèle, puis chargez-le dans la CloudFormation console pour créer une nouvelle CloudFormation pile. Pour plus d'informations, consultez la section Utilisation de la CloudFormation console dans le guide de AWS CloudFormation l'utilisateur.

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

Le modèle étant ouvert dans la AWS CloudFormation console, entrez les options suivantes. Notez que certaines options seront préremplies dans le modèle. Vous pouvez simplement les laisser comme valeurs par défaut.

- Sous Fournir un nom de pile
  - Sous Nom de la pile, entrez :

    ```
    getstarted-sg
    ```
- Sous Paramètres
  - Sous VpcId, choisissez le VPC dont le nom commence par. `hpc-networking`
  - (Facultatif) Sous ClientIpCidr, entrez une plage d'adresses IP plus restrictive pour le groupe de sécurité SSH entrant. Nous vous recommandons de limiter cela à votre propre adresse IP/sous-réseau (x.x.x.x/32 pour votre propre adresse IP ou x.x.x.x.x/24 pour la plage). Remplacez x.x.x.x par votre propre adresse IP PUBLIQUE. Vous pouvez obtenir votre adresse IP publique à l'aide d'outils tels que https://ifconfig.co/)

Surveillez l'état de la CloudFormation pile. Lorsqu'il atteint `CREATE_COMPLETE` le groupe de sécurité, les ressources sont prêtes.

Deux groupes de sécurité ont été créés, avec les noms suivants :

- `cluster-getstarted-sg`— il s'agit du groupe de sécurité du cluster
- `inbound-ssh-getstarted-sg`— il s'agit d'un groupe de sécurité permettant l'accès SSH entrant

Création d'un cluster dans AWS PCS

Dans AWS PCS, un cluster est une ressource permanente permettant de gérer les ressources et d'exécuter les charges de travail. Vous créez un cluster pour un planificateur spécifique (AWS PCS supporte actuellement Slurm) dans un sous-réseau d'un VPC nouveau ou existant. Le cluster accepte et planifie les tâches, et lance également les nœuds de calcul (EC2 instances) qui traitent ces tâches.

Pour créer votre cluster

1. Ouvrez la console AWS PCS et choisissez Create cluster.
2. Dans la section Détails du cluster, entrez les champs suivants :
   - Nom du cluster — Entrez `get-started`
   - Planificateur — Sélectionnez la version 25.05 de Slurm
   - Taille du contrôleur — Sélectionnez Petit
3. Dans la section Mise en réseau, sélectionnez des valeurs pour les champs suivants :
   - VPC — Choisissez le VPC nommé `hpc-networking:Large-Scale-HPC`
   - Sous-réseau : sélectionnez le sous-réseau dont le nom commence par `hpc-networking:PrivateSubnetA`
   - Groupes de sécurité : sélectionnez le groupe de sécurité du cluster nommé `cluster-getstarted-sg`
4. Choisissez Créer un cluster.

> **Note**
>
> Le champ État indique Création pendant le provisionnement du cluster. La création d'un cluster peut prendre plusieurs minutes.

Créez un espace de stockage partagé pour les AWS PC dans Amazon Elastic File System

Amazon Elastic File System (Amazon EFS) est un AWS service qui fournit un stockage de fichiers entièrement élastique sans serveur afin que vous puissiez partager des données de fichiers sans provisionner ni gérer la capacité et les performances de stockage. Pour plus d'informations, consultez Qu'est-ce qu'Amazon Elastic File System ? dans le guide de l'utilisateur d'Amazon Elastic File System.

Le cluster de démonstration AWS PCS utilise un système de fichiers EFS pour fournir un répertoire de base partagé entre les nœuds du cluster. Créez un système de fichiers EFS dans le même VPC que votre cluster.

Pour créer un système de fichiers Amazon EFS

1. Accédez à la console Amazon EFS.
2. Assurez-vous qu'il est réglé sur le même point que celui Région AWS où vous allez essayer AWS PCS.
3. Choisissez Create file system (Créer un système de fichiers).
4. Sur la page Créer un système de fichiers, définissez les paramètres suivants :
   - Pour Nom, saisissez `getstarted-efs`
   - Sous Virtual Private Cloud (VPC), choisissez le VPC nommé `hpc-networking:Large-Scale-HPC`
   - Sélectionnez Create (Créer). Cela vous renvoie à la page Systèmes de fichiers.
5. Notez l'ID du système de fichiers du système de `getstarted-efs` fichiers. Vous aurez besoin de ces informations ultérieurement.

Créez un espace de stockage partagé pour les AWS PC dans Amazon FSx for Lustre

Amazon FSx for Lustre permet de lancer et d'exécuter facilement et à moindre coût le système de fichiers Lustre, populaire et performant. Vous utilisez Lustre pour les charges de travail où la rapidité est importante, telles que l'apprentissage automatique, le calcul haute performance (HPC), le traitement vidéo et la modélisation financière. Pour plus d'informations, consultez Qu'est-ce qu'Amazon FSx pour Lustre ? dans le guide de l'utilisateur d'Amazon FSx for Lustre.

Le cluster de démonstration AWS PCS peut utiliser un système de fichiers FSx for Lustre pour fournir un répertoire partagé performant entre les nœuds du cluster. Créez un système de fichiers FSx for Lustre dans le même VPC que votre cluster.

Pour créer votre système de fichiers FSx for Lustre

1. Accédez à la FSx console Amazon.
2. Assurez-vous que la console est configurée pour utiliser la même chose Région AWS que votre cluster.
3. Choisissez Create file system (Créer un système de fichiers).

- Pour Sélectionner le type de système de fichiers, choisissez Amazon FSx pour Lustre, puis Next.

4. Sur la page Spécifier les détails du système de fichiers, définissez les paramètres suivants :

   - Sous Détails du système de fichiers
     - Pour Nom, saisissez `getstarted-fsx`
     - Pour le type de déploiement et de stockage, choisissez Persistent, SSD
     - Pour le débit par unité de stockage, choisissez 125 Mo/s/TiB
     - Pour Capacité de stockage, entrez 1,2 TiB
     - Pour la configuration des métadonnées, choisissez Automatique
     - Pour le type de compression des données, sélectionnez LZ4
   - Sous Réseau et sécurité
     - Pour Virtual Private Cloud (VPC), choisissez le VPC nommé `hpc-networking:Large-Scale-HPC`
     - Pour les groupes de sécurité VPC, laissez le groupe de sécurité nommé `default`
     - Pour Sous-réseau, choisissez le sous-réseau dont le nom commence par `hpc-networking:PrivateSubnetA`
   - Conservez les valeurs par défaut des autres options.
   - Choisissez Suivant.

5. Sur la page Réviser et créer, choisissez Créer un système de fichiers. Cela vous renvoie à la page Systèmes de fichiers.
6. Accédez à la page de détails du système de fichiers FSx for Lustre que vous avez créé.
7. Notez l'ID du système de fichiers et le nom du montage. Vous aurez besoin de ces informations ultérieurement.

> **Note**
>
> Le champ État indique Création pendant le provisionnement du système de fichiers. La création du système de fichiers peut prendre plusieurs minutes. Attendez qu'il soit terminé avant de poursuivre le reste du didacticiel.

Création de groupes de nœuds de calcul dans AWS PCS

Un groupe de nœuds de calcul est un ensemble virtuel de nœuds de calcul (instances EC2) que AWS PCS lance et gère. Lorsque vous définissez un groupe de nœuds de calcul, vous spécifiez des caractéristiques communes telles que les types d'instances EC2, le nombre d'instances minimal et maximal, les sous-réseaux VPC cibles, l'option d'achat préférée et la configuration de lancement personnalisée. AWS Le PCS lance, gère et arrête efficacement les nœuds de calcul d'un groupe de nœuds de calcul, conformément à ces paramètres. Le cluster de démonstration utilise un groupe de nœuds de calcul pour fournir des nœuds de connexion pour l'accès des utilisateurs, et un groupe de nœuds de calcul distinct pour traiter les tâches. Les rubriques suivantes décrivent les procédures permettant de configurer ces groupes de nœuds de calcul dans votre cluster.

Rubriques

- Création d'un profil d'instance pour AWS PCS
- Création de modèles de lancement pour AWS PCS
- Création d'un groupe de nœuds de calcul pour les nœuds de connexion dans AWS PCS
- Création d'un groupe de nœuds de calcul pour exécuter des tâches de calcul dans AWS PCS

Création d'un profil d'instance pour AWS PCS

Les groupes de nœuds de calcul nécessitent un profil d'instance lors de leur création. Si vous utilisez AWS Management Console pour créer un rôle pour Amazon EC2, la console crée automatiquement un profil d'instance et lui attribue le même nom que le rôle. Pour plus d'informations, consultez la section Utilisation des profils d'instance dans le Guide de Gestion des identités et des accès AWS l'utilisateur.

Dans la procédure suivante, vous utiliserez le AWS Management Console pour créer un rôle pour Amazon EC2, qui crée également le profil d'instance pour vos groupes de nœuds de calcul.

Pour créer le profil de rôle et d'instance

- Accédez à la Console IAM.
- Sous Access Management (Gestion des accès), choisissez Policies (politiques).
  - Choisissez Create Policy (Créer une politique).
  - Sous Spécifier les autorisations, dans Éditeur de politiques, sélectionnez JSON.
  - Remplacez le contenu de l'éditeur de texte par le suivant :

JSON

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

- Choisissez Suivant.
- Sous Vérifier et créer, dans le champ Nom de la politique, entrez`AWSPCS-getstarted-policy`.
- Choisissez Create Policy (Créer une politique).

- Sous Access Management (Gestion des accès), choisissez Roles (Rôles).
- Choisissez Créer un rôle.
- Sous Sélectionner une entité de confiance :
  - Pour le type d'entité de confiance, sélectionnez AWS service
  - Sous Cas d'utilisation, sélectionnez EC2.
    - Ensuite, sous Choisir un cas d'utilisation pour le service spécifié, choisissez EC2.
  - Choisissez Suivant.
- Sous Ajouter des autorisations :
  - Dans Politiques d'autorisations, recherchez AWSPCS-getstarted-policy.
  - Cochez la case à côté AWSPCS-getstartedde -policy pour l'ajouter au rôle.
  - Dans Politiques d'autorisations, recherchez Amazon SSMManaged InstanceCore.
  - Cochez la case à côté SSMManaged InstanceCore d'Amazon pour l'ajouter au rôle.
  - Choisissez Suivant.
- Sous Nom, passez en revue et créez :
  - Sous Détails du rôle :
    - Pour le Nom du rôle, saisissez `AWSPCS-getstarted-role`.

- Choisissez Créer un rôle.

Création de modèles de lancement pour AWS PCS

Lorsque vous créez un groupe de nœuds de calcul, vous fournissez un modèle de lancement EC2 que AWS PCS utilise pour configurer les instances EC2 qu'il lance. Cela inclut les paramètres tels que les groupes de sécurité et les scripts qui s'exécutent au lancement de l'instance.

Au cours de cette étape, un CloudFormation modèle sera utilisé pour créer deux modèles de lancement EC2. Un modèle sera utilisé pour créer des nœuds de connexion et l'autre pour créer des nœuds de calcul. La principale différence entre eux est que les nœuds de connexion peuvent être configurés pour autoriser l'accès SSH entrant.

Accédez au CloudFormation modèle

Utilisez l'URL suivante pour télécharger le CloudFormation modèle, puis chargez-le dans la CloudFormation console pour créer une nouvelle CloudFormation pile. Pour plus d'informations, consultez la section Utilisation de la CloudFormation console dans le guide de AWS CloudFormation l'utilisateur.

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

Utiliser le CloudFormation modèle pour créer des modèles de lancement EC2

Pour compléter le CloudFormation modèle dans la CloudFormation console, procédez comme suit :

- Sous Fournir un nom de pile :
  - Sous Nom de la pile, entrez`getstarted-lt`.
- Sous Paramètres :
  - Sous Sécurité
    - Pour VpcSecurityGroupId, sélectionnez le groupe de sécurité nommé `default` dans le VPC de votre cluster.
    - Pour ClusterSecurityGroupId, sélectionnez le groupe nommé `cluster-getstarted-sg`
    - Pour SshSecurityGroupId, sélectionnez le groupe nommé `inbound-ssh-getstarted-sg`
    - Pour SshKeyName, sélectionnez votre paire de clés SSH préférée.
  - Sous Systèmes de fichiers

  - Pour EfsFilesystemId, entrez l'ID du système de fichiers à partir du système de fichiers EFS que vous avez créé plus tôt dans le didacticiel.
  - Pour FSxLustreFilesystemId, entrez l'ID du système de fichiers à partir du système de fichiers FSx for Lustre que vous avez créé plus tôt dans le didacticiel.
  - Pour FSxLustreFilesystemMountName, entrez le même FSx nom de montage pour le système de fichiers Lustre.
- Choisissez Next, puis de nouveau Next.
- Sélectionnez Soumettre.

Surveillez l'état de la CloudFormation pile. Lorsqu'il atteint, `CREATE_COMPLETE` le modèle de lancement est prêt à être utilisé.

> **Note**
>
> Pour voir toutes les ressources créées par le CloudFormation modèle, ouvrez la CloudFormation console. Choisissez la pile `getstarted-lt`, puis choisissez l'onglet Ressources.

Création d'un groupe de nœuds de calcul pour les nœuds de connexion dans AWS PCS

Un groupe de nœuds de calcul est un ensemble virtuel de nœuds de calcul (instances EC2) que AWS PCS lance et gère. Lorsque vous définissez un groupe de nœuds de calcul, vous spécifiez des caractéristiques communes telles que les types d'instances EC2, le nombre d'instances minimal et maximal, les sous-réseaux VPC cibles, l'option d'achat préférée et la configuration de lancement personnalisée. AWS Le PCS lance, gère et arrête efficacement les nœuds de calcul d'un groupe de nœuds de calcul, conformément à ces paramètres.

Au cours de cette étape, vous allez lancer un groupe de nœuds de calcul statique qui fournit un accès interactif au cluster. Vous pouvez utiliser SSH ou Amazon EC2 Systems Manager (SSM) pour vous y connecter, puis exécuter des commandes shell et gérer les tâches Slurm.

Pour créer le groupe de nœuds de calcul

- Ouvrez la console AWS PCS et accédez à Clusters.

- Sélectionnez le cluster nommé `get-started`
- Accédez à Compute node groups et choisissez Create.
- Dans la section Configuration du groupe de nœuds de calcul, fournissez les informations suivantes :
  - Nom du groupe de nœuds de calcul — Entrez`login`.
- Sous Configuration informatique, entrez ou sélectionnez les valeurs suivantes :
  - Modèle de lancement EC2 — Choisissez le modèle de lancement dont le nom est `login-getstarted-lt`
  - Profil d'instance IAM — Choisissez le profil d'instance nommé `AWSPCS-getstarted-role`
  - Sous-réseaux : sélectionnez le sous-réseau dont le nom commence par. `hpc-networking:PublicSubnetA`
  - Instances — Sélectionnez`c6i.xlarge`.
  - Configuration de mise à l'échelle : pour le nombre minimal d'instances, entrez**1**. Pour Nombre maximal d'instances, entrez**1**.
- Sous Paramètres supplémentaires, spécifiez les éléments suivants :
  - ID AMI — Sélectionnez l'AMI que vous souhaitez utiliser, dont le nom est au format suivant :

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

    Pour plus d'informations sur cet exemple AMIs, consultezUtilisation d'exemples d'Amazon Machine Images (AMIs) avec AWS PCS.
- Choisissez Créer un groupe de nœuds de calcul.

Le champ Status indique Création pendant le provisionnement du groupe de nœuds de calcul. Vous pouvez passer à l'étape suivante du didacticiel pendant qu'il est en cours.

Création d'un groupe de nœuds de calcul pour exécuter des tâches de calcul dans AWS PCS

Au cours de cette étape, vous allez lancer un groupe de nœuds de calcul qui évolue de manière élastique pour exécuter les tâches soumises au cluster.

Pour créer le groupe de nœuds de calcul

- Ouvrez la console AWS PCS et accédez à Clusters.

- Sélectionnez le cluster nommé `get-started`
- Accédez à Compute node groups et choisissez Create.
- Dans la section Configuration du groupe de nœuds de calcul, fournissez les informations suivantes :
  - Nom du groupe de nœuds de calcul — Entrez`compute-1`.
- Sous Configuration informatique, entrez ou sélectionnez les valeurs suivantes :
  - Modèle de lancement EC2 — Choisissez le modèle de lancement dont le nom est `compute-getstarted-lt`
  - Profil d'instance IAM — Choisissez le profil d'instance nommé `AWSPCS-getstarted-role`
  - Sous-réseaux : sélectionnez le sous-réseau dont le nom commence par. `hpc-networking:PrivateSubnetA`
  - Instances — Sélectionnez`c6i.xlarge`.
  - Configuration de mise à l'échelle : pour le nombre minimal d'instances, entrez**`0`**. Pour Nombre maximal d'instances, entrez**`4`**.
- Sous Paramètres supplémentaires, spécifiez les éléments suivants :
  - ID AMI — Sélectionnez l'AMI que vous souhaitez utiliser, dont le nom est au format suivant :

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

    Pour plus d'informations sur cet exemple AMIs, consultezUtilisation d'exemples d'Amazon Machine Images (AMIs) avec AWS PCS.
- Choisissez Créer un groupe de nœuds de calcul.

Le champ Status indique Création pendant le provisionnement du groupe de nœuds de calcul.

> **Important**
>
> Attendez que le champ État indique Actif avant de passer à l'étape suivante de ce didacticiel.

Créez une file d'attente pour gérer les tâches dans AWS PCS

Vous soumettez une tâche à une file d'attente pour l'exécuter. La tâche reste dans la file d'attente jusqu'à ce que AWS PCS la planifie pour qu'elle s'exécute sur un groupe de nœuds de calcul.

Chaque file d'attente est associée à un ou plusieurs groupes de nœuds de calcul, qui fournissent les EC2 instances nécessaires pour effectuer le traitement.

Au cours de cette étape, vous allez créer une file d'attente qui utilise le groupe de nœuds de calcul pour traiter les tâches.

Pour créer une file d'attente

- Ouvrez la console AWS PCS.
- Sélectionnez le cluster nommé`get-started`.
- Accédez à Calculer les groupes de nœuds et assurez-vous que le statut du `compute-1` groupe est Actif.

> **Important**
>
> Le statut du `compute-1` groupe doit être Actif avant de passer à l'étape suivante.

- Accédez à Files d'attente et choisissez Créer une file d'attente.
  - Dans la section Configuration de la file d'attente, indiquez les valeurs suivantes :
    - Nom de la file d'attente — Entrez ce qui suit : `demo`
    - Groupes de nœuds de calcul : sélectionnez le groupe de nœuds de calcul nommé`compute-1`.
- Choisissez Créez une file d'attente.

Le champ État indique Création pendant la création de la file d'attente.

> **Important**
>
> Attendez que le champ État indique Actif avant de passer à l'étape suivante de ce didacticiel.

Connectez-vous à votre cluster AWS PCS

Lorsque le statut du groupe de nœuds de `login` calcul devient actif, vous pouvez vous connecter à l' EC2 instance qu'il a créée.

Pour vous connecter au nœud de connexion

- Ouvrez la console AWS PCS et accédez à Clusters.

- Sélectionnez le cluster nommé`get-started`.
- Choisissez Compute node groups.
- Accédez au groupe de nœuds de calcul nommé`login`.
- Trouvez l'ID du groupe de nœuds Compute.
- Dans une autre fenêtre ou un autre onglet du navigateur, ouvrez la EC2 console Amazon.
  - Choisissez Instances.
  - Recherchez des EC2 instances avec la balise suivante. Remplacez *`node-group-id`* par la valeur de l'ID du groupe de nœuds de calcul de l'étape précédente. Il devrait y avoir une instance.

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

  - Connectez-vous à l' EC2 instance. Vous pouvez utiliser le gestionnaire de session ou SSH.

    Session Manager

    - Sélectionnez l'instance.
    - Choisissez Se connecter.
    - Sous Connect to instance, sélectionnez Session Manager.
    - Choisissez Se connecter.
    - Choisissez Se connecter. Une borne interactive s'ouvre dans votre navigateur.

    SSH

    - Sélectionnez l'instance.
    - Choisissez Se connecter.
    - Sous Connect to instance, sélectionnez SSH client.
    - Suivez les instructions fournies par la console.

    > **Note**
    >
    > Le nom d'utilisateur de l'instance ne l'est **`ec2-user`**pas`root`.

Explorez l'environnement de cluster dans AWS PCS

Une fois connecté au cluster, vous pouvez exécuter des commandes shell. Par exemple, vous pouvez changer d'utilisateur, travailler avec des données sur des systèmes de fichiers partagés et interagir avec Slurm.

Changer d'utilisateur

Si vous vous êtes connecté au cluster à l'aide du gestionnaire de session, vous êtes peut-être connecté en tant que`ssm-user`. Il s'agit d'un utilisateur spécial créé pour le gestionnaire de session. Passez à l'utilisateur par défaut sur Amazon Linux 2 à l'aide de la commande suivante. Vous n'aurez pas besoin de le faire si vous vous êtes connecté via SSH.

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

Travailler avec des systèmes de fichiers partagés

Vous pouvez vérifier que le système de fichiers EFS et FSx les systèmes de fichiers Lustre sont disponibles à l'aide de la commande`df -h`. La sortie de votre cluster doit ressembler à ce qui suit :

```
[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
```

Le système de `/home` fichiers monte 127.0.0.1 et possède une très grande capacité. Il s'agit du système de fichiers EFS que vous avez créé plus tôt dans le didacticiel. Tous les fichiers écrits ici seront disponibles `/home` sur tous les nœuds du cluster.

Le système de `/shared` fichiers monte une adresse IP privée et a une capacité de 1,2 To. Il s'agit du système de fichiers FSx for Lustre que vous avez créé plus tôt dans le didacticiel. Tous les fichiers écrits ici seront disponibles `/shared` sur tous les nœuds du cluster.

Interagir avec Slurm

Rubriques

- Répertorier les files d'attente et les nœuds
- Afficher les offres d'emploi

Répertorier les files d'attente et les nœuds

Vous pouvez répertorier les files d'attente et les nœuds auxquels elles sont associées à l'aide`sinfo`. La sortie de votre cluster doit ressembler à ce qui suit :

```
[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 ~]$
```

Notez le nom de la partition`demo`. Son statut est `up` et il dispose d'un maximum de 4 nœuds. Il est associé aux nœuds du groupe de `compute-1` nœuds. Si vous modifiez le groupe de nœuds de calcul et augmentez le nombre maximum d'instances à 8, le nombre de nœuds sera lu 8 et la liste des nœuds sera lue`compute-1-[1-8]`. Si vous avez créé un deuxième groupe de nœuds de calcul nommé `test` avec 4 nœuds et que vous l'avez ajouté à la `demo` file d'attente, ces nœuds apparaîtront également dans la liste des nœuds.

Afficher les offres d'emploi

Vous pouvez répertorier toutes les tâches du système, quel que soit leur état, avec`squeue`. La sortie de votre cluster doit ressembler à ce qui suit :

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

Réessayez de l'exécuter `squeue` ultérieurement, lorsqu'une tâche Slurm est en attente ou en cours d'exécution.

Exécuter une tâche à nœud unique dans AWS PCS

Pour exécuter une tâche à l'aide de Slurm, vous devez préparer un script de soumission spécifiant les exigences de la tâche et le soumettre à une file d'attente à l'aide de la `sbatch` commande.

Généralement, cela se fait à partir d'un répertoire partagé, de sorte que les nœuds de connexion et de calcul disposent d'un espace commun pour accéder aux fichiers.

Connectez-vous au nœud de connexion de votre cluster et exécutez les commandes suivantes à l'invite du shell.

- Devenez l'utilisateur par défaut. Accédez au répertoire partagé.

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

- Utilisez les commandes suivantes pour créer un exemple de script de tâche :

```
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
```

- Soumettez le script de tâche au planificateur Slurm :

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

- Lorsque la tâche est soumise, elle renvoie un identifiant de tâche sous forme de numéro. Utilisez cet identifiant pour vérifier le statut du travail. Remplacez *job-id* dans la commande suivante par le numéro renvoyé par`sbatch`.

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

  Example

```
squeue --job 1
```

  La `squeue` commande renvoie un résultat similaire à ce qui suit :

```
JOBID PARTITION NAME USER     ST TIME NODES NODELIST(REASON)
```

```
1     demo      test ec2-user CF 0:47 1     compute-1
```

- Continuez à vérifier l'état de la tâche jusqu'à ce qu'elle atteigne le statut R (en cours). Le travail est terminé quand il `squeue` ne renvoie rien.
- Inspectez le contenu du `/shared` répertoire.

```
ls -alth /shared
```

Le résultat de la commande est similaire à ce qui suit :

```
-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
```

Les fichiers `single.1.err` ont été nommés `single.1.out` et écrits par l'un des nœuds de calcul de votre cluster. Comme la tâche a été exécutée dans un répertoire partagé (`/shared`), elles sont également disponibles sur votre nœud de connexion. C'est pourquoi vous avez configuré un système de fichiers FSx for Lustre pour ce cluster.

- Inspectez le contenu du `single.1.out` fichier.

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

La sortie est similaire à ce qui suit :

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

Exécuter une tâche MPI multi-nœuds avec Slurm sur PCS AWS

Ces instructions montrent comment utiliser Slurm pour exécuter une tâche MPI (Message Passing Interface) dans PCS. AWS

Exécutez les commandes suivantes à l'invite du shell de votre nœud de connexion.

- Devenez l'utilisateur par défaut. Accédez à son répertoire personnel.

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

```
cd ~/
```

- Créez du code source dans le langage de programmation 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
```

- Chargez le module OpenMPI.

```
module load openmpi
```

- Compilez le programme C.

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

- Rédigez un script de soumission de tâches 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
```

- Accédez au répertoire partagé.

```
cd /shared
```

- Soumettez le script de tâche.

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

- `squeue`À utiliser pour surveiller le travail jusqu'à ce qu'il soit terminé.
- Vérifiez le contenu de `multi.out` :

```
cat multi.out
```

La sortie est similaire à ce qui suit. Notez que chaque rang possède sa propre adresse IP car il s'est exécuté sur un nœud différent.

```
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
```

Supprimer vos AWS ressources pour AWS PCS

Une fois que vous avez terminé avec les groupes de clusters et de nœuds que vous avez créés pour ce didacticiel, vous devez supprimer les ressources que vous avez créées.

> **Important**
>
> Vous recevez des frais de facturation pour toutes les ressources utilisées dans votre Compte AWS

Pour supprimer les ressources AWS PCS que vous avez créées pour ce didacticiel

- Ouvrez la console AWS PCS.
- Accédez au cluster nommé get-started.
- Accédez à la section Files d'attente.
- Sélectionnez la file d'attente nommée demo.
- Sélectionnez Delete (Supprimer).

> **Important**
>
> Attendez que la file d'attente soit supprimée avant de continuer.

- Accédez à la section Groupes de nœuds de calcul.
- Sélectionnez le groupe de nœuds de calcul nommé compute-1.
- Sélectionnez Delete (Supprimer).
- Sélectionnez le groupe de nœuds de calcul nommé login.
- Sélectionnez Delete (Supprimer).

> **Important**
>
> Attendez que les deux groupes de nœuds de calcul aient été supprimés avant de continuer.

- Sur la page détaillée du cluster pour démarrer, choisissez Supprimer.

> **Important**
>
> Attendez que le cluster ait été supprimé avant de passer aux étapes suivantes.

Pour supprimer les autres AWS ressources que vous avez créées pour ce didacticiel

- Ouvrez la console IAM.
  - Sélectionnez Roles (Rôles).
  - Sélectionnez le rôle nommé AWSPCS-getstarted-role, puis choisissez Supprimer.
  - Une fois le rôle supprimé, choisissez Politiques.
  - Sélectionnez la politique nommée AWSPCS-getstarted-policy, puis choisissez Supprimer.
- Ouvrez la CloudFormation console.
  - Sélectionnez la pile nommée getstarted-lt.
  - Sélectionnez Delete (Supprimer).

> **Important**
>
> Attendez que la pile soit supprimée avant de continuer.

- Ouvrez la console Amazon EFS.
  - Choisissez File Systems (Systèmes de fichiers).
  - Sélectionnez le système de fichiers nommé getstarted-efs.
  - Sélectionnez Delete (Supprimer).

> **Important**
>
> Attendez que le système de fichiers soit supprimé avant de continuer.

- Ouvrez la FSx console Amazon.
  - Choisissez File Systems (Systèmes de fichiers).
  - Sélectionnez le système de fichiers nommé getstarted-fsx.
  - Sélectionnez Delete (Supprimer).

> **Important**
>
> Attendez que le système de fichiers soit supprimé avant de continuer.

- Ouvrez la CloudFormation console.
  - Sélectionnez la pile nommée getstarted-sg.
  - Sélectionnez Delete (Supprimer).
- Ouvrez la CloudFormation console.
  - Sélectionnez la pile nommée hpc-networking.
  - Sélectionnez Supprimer.

Commencez avec CloudFormation AWS PCS

Vous pouvez l'utiliser AWS CloudFormation pour créer un cluster AWS PCS. CloudFormation vous permet de créer et de provisionner des déploiements AWS d'infrastructure de manière prévisible et répétée. Vous pouvez l'utiliser CloudFormation pour provisionner automatiquement les ressources de nombreux AWS services afin de créer des applications hautement fiables, évolutives et rentables AWS Cloud sans créer ni configurer l' AWS infrastructure sous-jacente. CloudFormation vous permet d'utiliser un fichier modèle pour créer et supprimer un ensemble de ressources en une seule unité, appelée pile. Pour plus d'informations CloudFormation, voir Qu'est-ce que c'est CloudFormation ? dans le guide de AWS CloudFormation l'utilisateur. Pour plus d'informations sur les types de ressources AWS PCS dans CloudFormation, voir la référence des types de ressources AWS PCS dans le guide de AWS CloudFormation l'utilisateur.

Rubriques

- CloudFormation À utiliser pour créer un exemple de cluster AWS PCS
- Connectez-vous à un cluster AWS PCS créé avec CloudFormation
- Nettoyez un cluster AWS PCS dans CloudFormation
- Éléments d'un CloudFormation modèle pour AWS PCS
- CloudFormation modèles pour créer un exemple de cluster AWS PCS

CloudFormation À utiliser pour créer un exemple de cluster AWS PCS

La procédure suivante utilise un CloudFormation modèle dans le AWS Management Console pour créer un exemple de cluster AWS PCS. Pour plus d'informations CloudFormation, voir Qu'est-ce que c'est CloudFormation ? dans le guide de AWS CloudFormation l'utilisateur. Pour plus d'informations sur les types de ressources AWS PCS dans CloudFormation, voir la référence des types de ressources AWS PCS dans le guide de AWS CloudFormation l'utilisateur.

Pour créer le cluster d'échantillons

1. Choisissez le dans lequel Région AWS créer le cluster (le lien ouvre la CloudFormation console avec le modèle) :

    - USA Est (Virginie du Nord) (us-east-1)

- USA Est (Ohio) (us-east-2)
- USA Ouest (Oregon) (us-west-2)
- Asie-Pacifique (Singapour) (ap-southeast-1)
- Asie-Pacifique (Sydney) (ap-southeast-2)
- Asie-Pacifique (Tokyo) (ap-northeast-1)
- Europe (Francfort) (eu-central-1)
- Europe (Irlande) (eu-west-1)
- Europe (Londres) (eu-west-2)
- Europe (Stockholm) (eu-north-1)
- AWS GovCloud (USA Est) (us-gov-east-1)
- AWS GovCloud (US-Ouest) (us-gov-west-1)

2. Sous Fournir un nom de pile, entrez un nom descriptif. C'est le nom de votre CloudFormation pile. Le modèle utilise cette valeur comme nom pour votre cluster AWS PCS.
3. Sous Paramètres :

   a. Sous SlurmVersion, choisissez la version de Slurm que vous souhaitez que votre cluster utilise.

   b. Sous NodeArchitecture, choisissez x86 pour déployer un cluster qui utilise des instances compatibles x86_64, ou choisissez Graviton pour utiliser des instances Arm64.

   c. Pour KeyName, choisissez une paire de clés SSH pour accéder aux nœuds de connexion du cluster. Vérifiez que vous disposez du fichier PEM correspondant à la paire de clés que vous avez choisie.

   d. Pour ClientIpCidr, entrez une plage d'adresses IP au format CIDR pour contrôler l'accès aux nœuds de connexion.

   > ⚠ Warning
   >
   > La valeur par défaut de `0.0.0.0/0` autorise l'accès depuis toutes les adresses IP.

   e. Conservez les valeurs de HpcRecipesS3Bucket et de S3Bucket HpcRecipesBranchcomme valeurs par défaut.

4. Sous Capacités et transformations :

   a. Cochez la case pour confirmer que des ressources IAM CloudFormation seront créées.

b. Cochez la case pour confirmer que des ressources IAM CloudFormation seront créées avec des noms personnalisés.

c. Cochez la case `CAPABILITY_AUTO_EXPAND` pour accuser réception de la nouvelle pile. Pour plus d'informations, consultez CreateStack dans la Référence d'API AWS CloudFormation .

5. Sélectionnez Créer la pile.

6. Surveillez l'état de votre pile. Vous pouvez vous connecter au cluster une fois que l'état de la pile est atteint`CREATE_COMPLETE`.

Connectez-vous à un cluster AWS PCS créé avec CloudFormation

Après avoir créé un cluster AWS PCS à partir d'un CloudFormation modèle, vous pouvez utiliser la console AWS PCS (dans le AWS Management Console) pour administrer le cluster. Vous pouvez également vous connecter à l'un des nœuds de connexion du cluster pour administrer le cluster, exécuter des tâches et gérer les données. La CloudFormation pile fournit des liens que vous pouvez utiliser pour vous connecter à votre cluster.

Pour vous connecter à votre cluster

1. Ouvrez la console CloudFormation.

2. Choisissez la pile que vous avez créée.

3. Choisissez l'onglet Sorties de la pile.

   La pile fournit les liens suivants :

   - PcsConsoleUrl— Cliquez sur ce lien pour ouvrir la console AWS PCS avec le cluster sélectionné. Vous pouvez l'utiliser pour explorer les configurations du cluster, du groupe de nœuds et de la file d'attente.
   - Ec2 ConsoleUrl — Cliquez sur ce lien pour ouvrir la console Amazon EC2, filtrée pour afficher les instances gérées par le groupe de nœuds de connexion du cluster.

     Dans cette vue, vous pouvez sélectionner une instance et choisir Connect. L'instance du cluster d'échantillons prend en charge le SSH entrant et AWS Systems Manager les connexions dans un navigateur Web. Pour de plus amples informations, veuillez consulter Connectez-vous à votre cluster AWS PCS.

Après vous être connecté à une instance de connexion, vous pouvez suivre le didacticiel à l'adresseExplorez l'environnement de cluster dans AWS PCS.

Nettoyez un cluster AWS PCS dans CloudFormation

Si vous avez CloudFormation créé votre cluster AWS PCS, vous pouvez ouvrir la CloudFormation console et supprimer la pile pour supprimer le cluster et toutes les ressources associées.

> **Important**
>
> Pour le cluster d'exemple, si vous avez créé des groupes de nœuds de calcul ou des files d'attente supplémentaires dans votre cluster (en plus `compute-1` des groupes `login` et créés par le CloudFormation modèle d'exemple), vous devez utiliser la console AWS PCS ou AWS CLI supprimer ces ressources avant de supprimer la CloudFormation pile. Pour de plus amples informations, veuillez consulter Supprimer un cluster dans AWS PCS.

Éléments d'un CloudFormation modèle pour AWS PCS

Un CloudFormation modèle comporte une ou plusieurs sections qui répondent chacune à un objectif spécifique. CloudFormation définit le format, la syntaxe et le langage standard d'un modèle. Pour plus d'informations, consultez la section Utilisation des CloudFormation modèles dans le Guide de AWS CloudFormation l'utilisateur.

CloudFormation les modèles sont hautement personnalisables et leurs formats peuvent donc varier. Pour comprendre les éléments nécessaires d'un CloudFormation modèle pour créer un cluster AWS PCS, nous vous recommandons d'examiner l'exemple de modèle que nous fournissons pour créer un cluster d'échantillons. Cette rubrique décrit brièvement les sections de cet exemple de modèle.

> **Important**
>
> Les exemples de code présentés dans cette rubrique ne sont pas complets. La présence d'ellipses (`[...]`) indique qu'un code supplémentaire n'est pas affiché. Pour télécharger le CloudFormation modèle complet au format YAML, consultez. CloudFormation modèles pour créer un exemple de cluster AWS PCS

Table des matières

- En-tête
- Métadonnées
- Parameters
- Mappages
- Ressources
- Sorties

En-tête

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

`AWSTemplateFormatVersion`identifie la version du format du modèle à laquelle le modèle est conforme. Pour plus d'informations, consultez la syntaxe de version du format du CloudFormation modèle dans le guide de AWS CloudFormation l'utilisateur.

`Transform`spécifie une macro CloudFormation utilisée pour traiter le modèle. Pour plus d'informations, consultez la section Transformation du CloudFormation modèle dans le guide de AWS CloudFormation l'utilisateur. La `AWS::Serverless-2016-10-31` transformation permet CloudFormation de traiter un modèle écrit dans la syntaxe AWS Serverless Application Model (AWS SAM). Pour plus d'informations, consultez la section `AWS::ServerlessTransformation` dans le guide de AWS CloudFormation l'utilisateur.

Métadonnées

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

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

La `metadata` section d'un CloudFormation modèle fournit des informations sur le modèle lui-même. L'exemple de modèle crée un cluster de calcul haute performance (HPC) complet qui utilise le AWS PCS. La section des métadonnées de l'exemple de modèle déclare les paramètres qui contrôlent la manière dont CloudFormation lance (approvisionne) la pile correspondante. Certains paramètres contrôlent le choix de l'architecture (`NodeArchitecture`), la version de Slurm (`SlurmVersion`) et les contrôles d'accès (`KeyNameetClientIpCidr`).

Parameters

La `Parameters` section définit les paramètres personnalisés du modèle. CloudFormation utilise ces définitions de paramètres pour créer et valider le formulaire avec lequel vous interagissez lorsque vous lancez une pile à partir de ce modèle.

```
Parameters:

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

  SlurmVersion:
    Type: String
    Default: 25.05
    Description: Version of Slurm to use
    AllowedValues:
         - 24.11
         - 25.05
```

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

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

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

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

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

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

Mappages

La `Mappings` section définit des paires clé-valeur qui spécifient des valeurs en fonction de certaines conditions ou dépendances.

```
Mappings:

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

Ressources

La `Resources` section déclare les AWS ressources à provisionner et à configurer dans le cadre de la pile.

```
Resources:

  [...]
```

Le modèle fournit l'infrastructure du cluster d'échantillons en couches. Cela commence par `Networking` pour la configuration du VPC. Le stockage est assuré par deux systèmes : `EfsStorage` pour le stockage partagé et `FSxLStorage` pour le stockage à hautes performances. Le cluster principal est établi par`PCSCluster`.

```

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

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

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

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

  [...]

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

Pour les ressources de calcul, le modèle crée deux groupes de nœuds : `PCSNodeGroupLogin` pour un seul nœud de connexion et `PCSNodeGroupCompute` pour un maximum de quatre nœuds de

calcul. Ces groupes de nœuds sont pris en charge par `PCSInstanceProfile` les autorisations et, par `PCSLaunchTemplate` exemple, les configurations.

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

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

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

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

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

La planification des tâches est gérée via`PCSQueueCompute`.

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

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

La sélection des AMI s'effectue automatiquement via la fonction Lambda Pcs AMILookup Fn et les ressources associées.

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

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

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

Sorties

Le modèle génère l'identification et la gestion des clusters URLs via `ClusterIdPcsConsoleUrl`, et`Ec2ConsoleUrl`.

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

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

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

CloudFormation modèles pour créer un exemple de cluster AWS PCS

Région AWS nom	Région AWS	Afficher la source	Pile de lancement
USA Est (Virginie du Nord)	us-east-1	Télécharger YAML	
USA Est (Ohio)	us-east-2	Télécharger YAML	
USA Ouest (Oregon)	us-west-2	Télécharger YAML	
Asie-Pacifique (Singapour)	ap-southeast-1	Télécharger YAML	

Région AWS nom	Région AWS	Afficher la source	Pile de lancement
Asie-Pacifique (Sydney)	ap-southeast-2	Télécharger YAML	
Asie-Pacifique (Tokyo)	ap-northeast-1	Télécharger YAML	
Europe (Francfort)	eu-central-1	Télécharger YAML	
Europe (Irlande)	eu-west-1	Télécharger YAML	
Europe (Londres)	eu-west-2	Télécharger YAML	
Europe (Stockholm)	eu-north-1	Télécharger YAML	
AWS GovCloud (USA Est)	us-gov-east-1	Télécharger YAML	
AWS GovCloud (US-Ouest)	us-gov-west-1	Télécharger YAML	

AWS Clusters PCS

Un cluster AWS PCS comprend les composants suivants :

- Instances gérées du logiciel de planification du système HPC, telles que le démon de contrôle Slurm (). `slurmctld`
- Composants qui s'intègrent au planificateur du système HPC pour approvisionner et gérer les instances Amazon. EC2
- Composants qui s'intègrent au planificateur du système HPC pour transmettre les journaux et les métriques à Amazon. CloudWatch

Ces composants s'exécutent dans un compte géré par AWS. Ils travaillent ensemble pour gérer les EC2 instances Amazon de votre compte client. AWS PCS fournit des interfaces réseau élastiques dans votre sous-réseau Amazon VPC pour fournir une connectivité entre le logiciel de planification et les instances EC2 Amazon (par exemple, pour permettre la planification de tâches par lots sur celles-ci et permettre aux utilisateurs d'exécuter des commandes du planificateur pour répertorier et gérer ces tâches).

Rubriques

- Création d'un cluster dans AWS PCS
- Mettre à jour un cluster dans AWS PCS
- Supprimer un cluster dans AWS PCS
- Taille du cluster en AWS PCS
- Utilisation des secrets de cluster dans AWS PCS

Création d'un cluster dans AWS PCS

Cette rubrique fournit une vue d'ensemble des options disponibles et décrit les éléments à prendre en compte lors de la création d'un cluster dans AWS Parallel Computing Service (AWS PCS). Si c'est la première fois que vous créez un cluster AWS PCS, nous vous recommandons de suivreCommencez avec AWS Parallel Computing Service. Le didacticiel peut vous aider à créer un système HPC fonctionnel sans étendre toutes les options disponibles et les architectures système possibles.

> **Note**
>
> Après avoir créé un cluster, vous pouvez modifier de nombreux paramètres de configuration sans avoir à reconstruire votre infrastructure. Pour de plus amples informations, veuillez consulter Mettre à jour un cluster dans AWS PCS.

> **Note**
>
> Vous pouvez configurer des paramètres Slurm personnalisés pour implémenter des politiques de planification avancées et une gestion des ressources. Pour de plus amples informations, veuillez consulter Configuration des paramètres personnalisés de Slurm dans PCS AWS.

Conditions préalables

- Un VPC et un sous-réseau existants qui répondent aux exigences. AWS Mise en réseau PCS Avant de déployer un cluster pour une utilisation en production, nous vous recommandons de bien connaître les exigences du VPC et du sous-réseau. Pour créer un VPC et un sous-réseau, consultez. Création d'un VPC pour votre AWS cluster PCS
- Un directeur IAM autorisé à créer et à gérer des ressources AWS PCS. Pour de plus amples informations, veuillez consulter Identity and Access Management pour le service de calcul AWS parallèle.

Création d'un cluster AWS PCS

Vous pouvez utiliser le AWS Management Console ou AWS CLI pour créer un cluster.

AWS Management Console

Pour créer un cluster

1. Ouvrez la console AWS PCS à l'adresse https://console.aws.amazon.com/pcs/home#/clusters et choisissez Create cluster.
2. Dans la section Configuration du cluster, entrez les champs suivants :

- Nom du cluster : nom de votre cluster. Un nom ne peut contenir que des caractères alphanumériques (sensibles à la casse) et des traits d'union. Il doit commencer par un caractère alphabétique et ne doit pas comporter plus de 40 caractères. Le nom doit être unique dans le Région AWS et dans Compte AWS lequel vous créez le cluster.
- Planificateur : choisissez un planificateur et une version. Pour de plus amples informations, veuillez consulter Versions Slurm en PCS AWS.
- Taille de la manette — Choisissez une taille pour votre manette. Cela détermine le nombre de tâches simultanées et de nœuds de calcul pouvant être gérés par le cluster AWS PCS. Vous ne pouvez définir la taille du contrôleur que lorsque le cluster est créé. Pour plus d'informations sur le dimensionnement, voirTaille du cluster en AWS PCS.

3. Dans la section Mise en réseau, sélectionnez des valeurs pour les champs suivants :

- Type de réseau — Choisissez le type d'adresse IP de votre cluster. Votre cluster peut utiliser l'un IPv4 ou l'autre IPv6, mais pas les deux. Le VPC et les sous-réseaux doivent utiliser le même type d'adresse réseau. Le bloc d'adresses IP que vous utilisez pour chaque sous-réseau doit comporter au moins une adresse disponible. AWS réserve certaines adresses de chaque sous-réseau. Pour plus d'informations, consultez Blocs d'adresse CIDR de sous-réseau dans le Guide de l'utilisateur Amazon VPC.
- VPC — Choisissez un VPC existant qui répond aux exigences du PCS. AWS Pour de plus amples informations, veuillez consulter AWS Exigences et considérations relatives au PCS, au VPC et aux sous-réseaux. Après avoir créé le cluster, vous ne pouvez pas modifier son VPC. Si aucun VPCs n'est répertorié, vous devez d'abord en créer un.
- Sous-réseau : tous les sous-réseaux disponibles dans le VPC sélectionné sont répertoriés. Choisissez un sous-réseau qui répond aux exigences du sous-réseau AWS PCS. Pour de plus amples informations, veuillez consulter AWS Exigences et considérations relatives au PCS, au VPC et aux sous-réseaux. Nous vous recommandons de sélectionner un sous-réseau privé pour éviter d'exposer les points de terminaison de votre planificateur à l'Internet public.
- Groupes de sécurité — Spécifiez le ou les groupes de sécurité que vous souhaitez que AWS PCS associe aux interfaces réseau qu'il crée pour votre cluster. Vous devez sélectionner au moins un groupe de sécurité qui autorise la communication entre votre cluster et ses nœuds de calcul. Vous pouvez sélectionner Création rapide d'un groupe de sécurité pour que AWS PCS en crée un avec la configuration nécessaire dans le VPC sélectionné, ou sélectionner un groupe de sécurité existant. Pour de plus amples

informations, veuillez consulter Exigences et considérations relatives aux groupes de sécurité.

4. (Facultatif) Dans la section de configuration de la comptabilité Slurm, vous pouvez activer la comptabilité Slurm et définir les paramètres de comptabilité. Pour de plus amples informations, veuillez consulter Comptabilité Slurm dans PCS AWS.
5. (Facultatif) Dans la section Configuration de Slurm, vous pouvez ajouter des paires de nom et de valeur des paramètres pour configurer des paramètres Slurm supplémentaires. Pour obtenir la liste complète des paramètres pris en charge, consultezParamètres Slurm personnalisés pour les clusters PCS AWS.
6. (Facultatif) Sous Balises, ajoutez des balises à votre cluster AWS PCS.
7. Choisissez Créer un cluster. Le champ Status s'affiche `Creating` lorsque le AWS PCS crée le cluster. Ce processus peut prendre plusieurs minutes.

> **Important**
>
> Il ne peut y avoir qu'un seul cluster Région AWS par `Creating` état Compte AWS. AWS PCS renvoie une erreur s'il existe déjà un cluster dans un `Creating` état lorsque vous essayez de créer un cluster.

AWS CLI

Pour créer un cluster

1. Créez votre cluster à l'aide de la commande suivante. Avant d'exécuter la commande, effectuez les remplacements suivants :
   - *`region`*Remplacez-le par l'ID dans Région AWS lequel vous souhaitez créer votre cluster, tel que`us-east-1`.
   - Remplacez *`my-cluster`* par un nom pour votre cluster. Un nom ne peut contenir que des caractères alphanumériques (sensibles à la casse) et des traits d'union. Il doit commencer par un caractère alphabétique et ne doit pas comporter plus de 40 caractères. Le nom doit être unique dans le cluster Région AWS et dans Compte AWS lequel vous créez le cluster.
   - *`25.05`*Remplacez-le par n'importe quelle version compatible de Slurm.

> **Note**
>
> AWS PCS prend actuellement en charge Slurm 25.05 et 24.11.

- Remplacez-le *SMALL* par n'importe quelle taille de cluster prise en charge. Cela détermine le nombre de tâches simultanées et de nœuds de calcul pouvant être gérés par le cluster AWS PCS. Il ne peut être défini que lors de la création du cluster. Pour plus d'informations sur le dimensionnement, voirTaille du cluster en AWS PCS.
- Remplacez la valeur de `subnetIds` par la vôtre. Nous vous recommandons de sélectionner un sous-réseau privé pour éviter d'exposer les points de terminaison de votre planificateur à l'Internet public.
- Spécifiez `securityGroupIds` ce que vous souhaitez que le AWS PCS associe aux interfaces réseau qu'il crée pour votre cluster. Les groupes de sécurité doivent se trouver dans le même VPC que le cluster. Vous devez sélectionner au moins un groupe de sécurité qui autorise la communication entre votre cluster et ses nœuds de calcul. Pour de plus amples informations, veuillez consulter Exigences et considérations relatives aux groupes de sécurité.

```
aws pcs create-cluster --region region \
    --cluster-name my-cluster \
    --scheduler type=SLURM,version=25.05 \
    --size SMALL \
    --networking subnetIds=subnet-ExampleId1,securityGroupIds=sg-ExampleId1
```

- pour l'utiliser IPv6, ajoutez-le `networkType=IPV6` à la `--networking` configuration.

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

- Vous pouvez éventuellement ajouter l'`--slurm-configration`option permettant de personnaliser le comportement de Slurm et de spécifier les options de configuration de Slurm. L'exemple suivant définit le temps d'inactivité réduit à 60 minutes (3 600 secondes), active la comptabilité Slurm et spécifie `slurm.conf` les paramètres comme valeur pour. `slurmCustomSettings` Pour de plus amples informations, veuillez consulter Comptabilité Slurm dans PCS AWS.

> **Note**
>
> La comptabilité est prise en charge pour Slurm 24.11 ou version ultérieure.

```
aws pcs create-cluster --region region \
    --cluster-name my-cluster \
    --scheduler type=SLURM,version=25.05 \
    --size SMALL \
    --networking subnetIds=subnet-ExampleId1,securityGroupIds=sg-ExampleId1
    --slurm-configuration
 scaleDownIdleTimeInSeconds=3600,accounting='{mode=STANDARD}',slurmCustomSettings='[{p
```

2. Le provisionnement du cluster peut prendre plusieurs minutes. Vous pouvez vérifier le statut de votre cluster avec la commande suivante. Ne créez pas de files d'attente ou de groupes de nœuds de calcul tant que le champ d'état du cluster n'est `ACTIVE` pas indiqué.

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

> **Important**
>
> Il ne peut y avoir qu'un seul cluster Région AWS par `Creating` état Compte AWS. AWS PCS renvoie une erreur s'il existe déjà un cluster dans un `Creating` état lorsque vous essayez de créer un cluster.

Prochaines étapes recommandées pour votre cluster

- Ajoutez des groupes de nœuds de calcul.
- Ajoutez des files d'attente.
- Activez la journalisation

Mettre à jour un cluster dans AWS PCS

AWS PCS vous permet de mettre à jour les configurations de cluster après leur création via l' UpdateCluster API ou la console. Vous pouvez modifier les paramètres du cluster sans reconstruire votre infrastructure, ce qui réduit les frais opérationnels et les interruptions.

Avantages des mises à jour du cluster

La mise à jour des clusters AWS PCS vous permet d'adapter l'infrastructure HPC aux nouvelles exigences sans interruption de service. Les modifications de configuration prennent quelques minutes au lieu des heures ou plus nécessaires à la reconstruction des clusters. Cette fonctionnalité est importante pour les environnements de production qui nécessitent des temps d'arrêt minimaux et pour les équipes qui doivent ajuster les paramètres du cluster en fonction de l'évolution des modèles de charge de travail.

Modifications de configuration prises en charge

Vous pouvez modifier trois catégories principales de paramètres :

- Configuration de la comptabilité : activez ou désactivez la comptabilité gérée et configurez les paramètres de rétention.
- Comportement réduit : ajustez le `scaleDownIdleTime` paramètre, qui contrôle la durée pendant laquelle les instances dynamiques restent inactives avant que AWS PCS ne les arrête automatiquement.
- Paramètres personnalisés de Slurm - Modifiez tous les paramètres Slurm pris en charge qui s'appliquent au niveau du cluster, notamment Prolog, Epilog et. SelectTypeParameters

Limites

Vous ne pouvez pas modifier certaines configurations après la création du cluster. Il s’agit des licences suivantes :

- Configurations des groupes de sécurité
- Sélection du sous-réseau VPC
- Taille du cluster
- Version Slurm
- Nom du cluster

Ces paramètres sont fondamentaux pour l'architecture du cluster et nécessitent la création d'un nouveau cluster pour les modifier.

Conditions préalables pour les mises à jour du cluster

Avant de mettre à jour un cluster, assurez-vous que les conditions suivantes sont remplies :

- Le cluster doit être en `ACTIVE`, `UPDATE_FAILED`, ou en `SUSPENDED` état
- Toutes les ressources associées (files d'attente, groupes de nœuds de calcul) doivent être en état `ACTIVE`
- Vous devez disposer des autorisations IAM appropriées pour l'opération UpdateCluster
- Aucune autre opération de mise à jour ne peut être en cours

Processus de mise à jour et impact sur le travail

Lors d'une opération de mise à jour, les nœuds de calcul continuent d'exécuter les tâches existantes même lorsque le contrôleur de cluster devient temporairement inaccessible. Cependant, le système ne peut pas accepter de nouvelles offres d'emploi ni prendre de décisions de planification pendant cette période.

Vous pouvez surveiller les mises à jour du cluster via la console et les interfaces API. Le cluster passera par les états suivants lors d'une mise à jour :

- `UPDATING`- Mise à jour en cours
- `ACTIVE`- Mise à jour terminée avec succès
- `UPDATE_FAILED`- La mise à jour a rencontré une erreur

Facturation lors des mises à jour

Les frais horaires standard pour votre cluster AWS PCS sont maintenus pendant les opérations de mise à jour. Lorsque vous mettez à jour un cluster pour désactiver la comptabilité, la facturation de la fonctionnalité de comptabilité s'arrête dès que le cluster entre dans l'`UPDATING`état. Lors de l'activation de la comptabilité, la facturation ne commence pas tant que le cluster n'a pas terminé la mise à jour avec succès et n'est pas revenu à l'`ACTIVE`état actuel.

Rubriques

- Mettre à jour un cluster AWS PCS
- Questions fréquemment posées sur la mise à jour des clusters dans AWS PCS
- Dépannage AWS des mises à jour du cluster

Mettre à jour un cluster AWS PCS

Suivez ces étapes pour modifier les paramètres du planificateur, la configuration de la comptabilité et les paramètres personnalisés de Slurm sur votre cluster. Pour de plus amples informations, veuillez consulter Paramètres Slurm personnalisés pour les clusters PCS AWS.

Prérequis

- Le cluster doit être en `ACTIVEUPDATE_FAILED`, ou en `SUSPENDED` état
- Toutes les ressources associées (files d'attente, groupes de nœuds de calcul) doivent être en état `ACTIVE`
- Aucune autre opération de mise à jour ne peut être en cours

Procédure

AWS Management Console

1. Ouvrez la console AWS PCS à l'adresse https://console.aws.amazon.com/pcs/.
2. Dans le panneau de navigation, choisissez Clusters.
3. Sélectionnez le cluster à mettre à jour.
4. Choisissez Modifier.
5. Sur la page Modifier le cluster, modifiez les paramètres souhaités :
   - Dans Configuration du planificateur, mettez à jour le temps d'inactivité réduit pour contrôler la durée pendant laquelle les instances dynamiques restent inactives avant leur arrêt automatique.
   - Modifiez les paramètres de type Prolog, Epilog et Select selon vos besoins.
   - Activez, désactivez ou configurez le temps de rétention pour la comptabilité gérée.
   - Sous Paramètres supplémentaires du planificateur, ajoutez, modifiez ou supprimez les paramètres personnalisés de Slurm. Pour plus d'informations sur les paramètres pris en charge, consultezParamètres Slurm personnalisés pour les clusters PCS AWS.

> **Note**
>
> Les champs qui ne peuvent pas être modifiés sont affichés en lecture seule et affichent leurs valeurs actuelles.

6. Choisissez Mettre à jour pour soumettre les modifications.
7. Surveillez l'état du cluster, qui s'affiche comme « Mise à jour » pendant le processus. Le statut change lorsque la mise à jour est terminée avec succès.

AWS CLI

1. Ouvrez un terminal ou une invite de commande.
2. Vérifiez l'état du cluster à l'aide de la commande suivante :

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

3. Soumettez une demande de mise à jour à l'aide de l'un des exemples suivants :

   - Pour activer la comptabilité gérée :

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

   - Pour mettre à jour un paramètre de Slurm Prolog :

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

   - Pour mettre à jour le temps d'inactivité réduit :

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

4. Surveillez la progression des mises à jour en vérifiant l'état du cluster :

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

Après une demande de mise à jour réussie, la commande renvoie l'objet Cluster avec toutes les modifications. L'état du cluster passe de `UPDATING` à une `ACTIVE` fois terminé.

Questions fréquemment posées sur la mise à jour des clusters dans AWS PCS

Obtenez des réponses aux questions les plus fréquemment posées sur la mise à jour des configurations de cluster dans AWS PCS.

Quels sont les paramètres que je peux modifier ?

Vous pouvez modifier la configuration comptable (activer/désactiver la comptabilité gérée), le comportement de réduction (paramètre scaleDownIdle Time) et tous les paramètres personnalisés pris en charge par Slurm qui s'appliquent au niveau du cluster. Vous ne pouvez pas modifier les groupes de sécurité, les sous-réseaux VPC, la taille du cluster, la version de Slurm ou le nom du cluster.

Puis-je mettre plusieurs mises à jour en file d'attente ?

Non Vous devez attendre que le cluster revienne à son `ACTIVE` état initial avant de soumettre une autre mise à jour. Toutes les ressources associées (files d'attente, groupes de nœuds de calcul) doivent également être en `ACTIVE` état.

Puis-je annuler une opération de mise à jour du cluster ?

Non, vous ne pouvez pas annuler une opération de mise à jour du cluster en cours.

Puis-je soumettre des tâches pendant que mon cluster est en cours de mise à jour ?

Nous vous recommandons d'éviter de soumettre des tâches lors des mises à jour du cluster. Il est possible que le contrôleur Slurm ne soit pas disponible pendant le processus de mise à jour.

Mes tâches continueront-elles à s'exécuter pendant les mises à jour du cluster ?

Oui, les tâches en cours continuent de s'exécuter sur les nœuds de calcul même lorsque le contrôleur de cluster devient brièvement inaccessible pendant le processus de mise à jour. Cependant, le statut de la tâche peut ne pas être mis à jour tant que le contrôleur ne sera pas de nouveau disponible.

Comment la facturation est-elle affectée lors des mises à jour ?

Les frais horaires standard sont maintenus pendant les opérations de mise à jour. Lors de la désactivation de la comptabilité, la facturation s'arrête lorsque le cluster entre en `UPDATING`

état. Lors de l'activation de la comptabilité, la facturation commence lorsque le cluster revient à `ACTIVE` l'état avec succès.

Dépannage AWS des mises à jour du cluster

Cette rubrique vous aide à identifier et à résoudre les problèmes courants qui peuvent survenir lors de la mise à jour des configurations de cluster.

La mise à jour échoue en raison d'une erreur de configuration comptable

Cause courante

Le cluster entre en `UPDATE_FAILED` état et le message d'erreur indique un problème de configuration comptable. Cela se produit généralement lorsque la configuration de gestion des comptes est incompatible avec la version actuelle de Slurm ou contient des paramètres non valides.

Résolution

Vérifiez la compatibilité de vos paramètres de comptabilité avec la version Slurm de votre cluster et soumettez une demande de mise à jour corrigée avec des paramètres de configuration valides.

La mise à jour échoue avec une erreur de paramètres personnalisés

Cause courante

Le cluster entre en `UPDATE_FAILED` état et le message d'erreur indique un problème de paramètres personnalisés de Slurm. Cela se produit lorsque vous fournissez des valeurs de paramètres Slurm non valides ou des combinaisons de paramètres non prises en charge.

Résolution

Validez vos paramètres personnalisés Slurm par rapport aux paramètres pris en charge et soumettez une demande de mise à jour corrigée avec des valeurs et des combinaisons de paramètres valides.

Impossible de soumettre une demande de mise à jour

Cause courante

Le bouton de mise à jour est désactivé dans la console ou l'API renvoie une erreur de niveau 400. Cela se produit lorsque le cluster n'est pas dans un état approprié, que les ressources associées ne sont pas actives ou que votre configuration présente des échecs de validation.

Résolution

Attendez que le cluster et toutes les ressources associées atteignent leur `ACTIVE` état, puis vérifiez que votre configuration ne comporte aucune erreur de validation avant de soumettre à nouveau la demande de mise à jour.

Erreurs de validation

Cause courante

La commande revient immédiatement avec une erreur HTTP de niveau 400 et un message descriptif. Cela se produit en raison d'un état du cluster, d'un état de ressource ou de paramètres de configuration non valides.

Résolution

Corrigez l'erreur de validation spécifique mentionnée dans la réponse et réessayez l'opération de mise à jour.

Supprimer un cluster dans AWS PCS

Cette rubrique explique comment supprimer un cluster AWS PCS.

Considérations relatives à la suppression d'un cluster AWS PCS

- Toutes les files d'attente associées au cluster doivent être supprimées pour que le cluster puisse être supprimé. Pour de plus amples informations, veuillez consulter Supprimer une file d'attente dans AWS PCS.
- Tous les groupes de nœuds de calcul associés au cluster doivent être supprimés pour que le cluster puisse être supprimé. Pour de plus amples informations, veuillez consulter Suppression d'un groupe de nœuds de calcul dans AWS PCS.

Supprimer le cluster

Vous pouvez utiliser le AWS Management Console ou AWS CLI pour supprimer un cluster.

AWS Management Console

Pour supprimer un cluster

1. Ouvrez la console AWS PCS.
2. Sélectionnez le cluster à supprimer.
3. Sélectionnez Delete (Supprimer).
4. Le champ État du cluster s'affiche`Deleting`. Cela peut prendre plusieurs minutes.

AWS CLI

Pour supprimer un cluster

1. Utilisez la commande suivante pour supprimer un cluster, avec les remplacements suivants :
   - *`region-code`*Remplacez-le par celui dans lequel se trouve Région AWS votre cluster.
   - Remplacez *`my-cluster`* par le nom ou l'ID de votre cluster.

   ```
   aws pcs delete-cluster --region region-code --cluster-identifier my-cluster
   ```
2. La suppression du cluster peut prendre plusieurs minutes. Vous pouvez vérifier l'état de votre cluster à l'aide de la commande suivante.

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

Taille du cluster en AWS PCS

AWS Le PCS fournit des clusters sécurisés et hautement disponibles, tout en automatisant les tâches clés telles que l'application de correctifs, le provisionnement des nœuds et les mises à jour.

Lorsque vous créez un cluster, vous sélectionnez sa taille en fonction de deux facteurs :

- Le nombre de nœuds de calcul qu'il gérera
- Le nombre de tâches actives et en file d'attente que vous comptez exécuter sur le cluster

> **Important**
>
> Vous ne pouvez pas modifier la taille du cluster après l'avoir créé. Si vous devez modifier la taille, vous devez créer un nouveau cluster.

Taille du cluster Slurm	Nombre d'instances gérées	Nombre de tâches actives et en attente
Petit	Jusqu'à 32	Jusqu'à 256
Moyenne	Jusqu'à 512	Jusqu'à 8192
Grand	Jusqu'à 2048	Jusqu'à 16384

Exemples

- Si votre cluster doit comporter jusqu'à 24 instances gérées et exécuter jusqu'à 100 tâches, choisissez Small.
- Si votre cluster doit comporter jusqu'à 24 instances gérées et exécuter jusqu'à 1 000 tâches, choisissez Medium.
- Si votre cluster doit comporter jusqu'à 1 000 instances gérées et exécuter jusqu'à 100 tâches, choisissez Large.
- Si votre cluster doit comporter jusqu'à 1 000 instances gérées et exécuter jusqu'à 10 000 tâches, choisissez Large.

Utilisation des secrets de cluster dans AWS PCS

Dans le cadre de la création d'un cluster, AWS PCS crée un secret de cluster qui est nécessaire pour se connecter au planificateur de tâches du cluster. Vous créez également des groupes de nœuds de calcul AWS PCS, qui définissent des ensembles d'instances à lancer en réponse à des événements de dimensionnement. AWS PCS configure les instances lancées par ces groupes de nœuds de calcul avec le secret du cluster afin qu'elles puissent se connecter au planificateur de tâches. Dans certains cas, vous souhaiterez peut-être configurer les clients Slurm manuellement. Les exemples incluent la création d'un nœud de connexion permanent ou la configuration d'un gestionnaire de flux de travail doté de fonctionnalités de gestion des tâches.

AWS PCS stocke le secret du cluster en tant que secret géré avec le préfixe `pcs!` in AWS Secrets Manager. Le coût du secret est inclus dans les frais d'utilisation du AWS PCS. Vous pouvez alterner les secrets du cluster AWS Secrets Manager afin de garantir la conformité en matière de sécurité et de remédier aux risques de sécurité potentiels.

Rubriques

- AWS Secrets Manager À utiliser pour trouver le secret du cluster
- Utilisez AWS PCS pour trouver le secret du cluster
- Obtenez le secret du cluster Slurm
- Rotation des secrets de cluster dans AWS PCS

AWS Secrets Manager À utiliser pour trouver le secret du cluster

AWS Management Console

1. Accédez à la console Secrets Manager.
2. Choisissez Secrets, puis recherchez le `pcs!` préfixe.

> **Note**
>
> Un secret de cluster AWS PCS porte un nom sous la forme `pcs!slurm-secret-`*`cluster-id`* où se *`cluster-id`* trouve l'ID de cluster AWS PCS.

AWS CLI

Chaque secret de cluster AWS PCS est également étiqueté avec`aws:pcs:`*`cluster-id`*. Vous pouvez obtenir l'ID secret d'un cluster à l'aide de la commande suivante. Effectuez les substitutions suivantes avant d'exécuter la commande :

- *`region`*Remplacez-le par le Région AWS pour créer votre cluster dans, par exemple`us-east-1`.
- Remplacez *`cluster-id`* par l'ID du cluster AWS PCS pour lequel vous souhaitez trouver le secret du cluster.

```
aws secretsmanager list-secrets \
```

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

Utilisez AWS PCS pour trouver le secret du cluster

Vous pouvez utiliser le AWS CLI pour trouver l'ARN d'un secret de cluster AWS PCS. Entrez la commande suivante en effectuant les substitutions suivantes :

- *region*Remplacez-le par le Région AWS pour créer votre cluster dans, par exemple`us-east-1`.
- Remplacez *my-cluster* par le nom ou l'identifiant de votre cluster.

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

L'exemple de sortie suivant provient de la `get-cluster` commande. Vous pouvez utiliser `secretArn` et `secretVersion` ensemble pour obtenir le secret.

```
{
    "cluster": {
        "name": "get-started",
        "id": "pcs_123456abcd",
        "arn": "arn:aws:pcs:us-east-1:111122223333:cluster/pcs_123456abcd",
        "status": "ACTIVE",
        "createdAt": "2024-12-17T21:03:52+00:00",
        "modifiedAt": "2024-12-17T21:03:52+00:00",
        "scheduler": {
            "type": "SLURM",
            "version": "25.05"
        },
        "size": "SMALL",
        "slurmConfiguration": {
            "authKey": {
                "secretArn": "arn:aws:secretsmanager:us-east-1:111122223333:secret:pcs!
slurm-secret-pcs_123456abcd-a12ABC",
                "secretVersion": "ef232370-d3e7-434c-9a87-ec35c1987f75"
            }
        },
        "networking": {
            "subnetIds": [
                "subnet-0123456789abcdef0"
```

```
            ],
            "securityGroupIds": [
                "sg-0123456789abcdef0"
            ]
        },
        "endpoints": [
            {
                "type": "SLURMCTLD",
                "privateIpAddress": "10.3.149.220",
                "port": "6817"
            }
        ]
    }
}
```

Obtenez le secret du cluster Slurm

Vous pouvez utiliser Secrets Manager pour obtenir la version actuelle codée en base64 d'un secret de cluster Slurm. L'exemple suivant utilise le. AWS CLI Effectuez les substitutions suivantes avant d'exécuter la commande.

- *region*Remplacez-le par le Région AWS pour créer votre cluster dans, par exemple`us-east-1`.
- *secret-arn*Remplacez-le par le `secretArn` provenant d'un cluster AWS PCS.

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

Pour plus d'informations sur l'utilisation du secret du cluster Slurm, consultez. Utilisation d'instances autonomes comme nœuds de connexion AWS PCS

Permissions

Vous utilisez un principal IAM pour obtenir le secret du cluster Slurm. Le directeur de l'IAM doit être autorisé à lire le secret. Pour plus d'informations, consultez la section Termes et concepts relatifs aux rôles dans le guide de Gestion des identités et des accès AWS l'utilisateur.

L'exemple de politique IAM suivant autorise l'accès à un exemple de secret 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"
        }
    ]
}
```

Rotation des secrets de cluster dans AWS PCS

Utilisez AWS Secrets Manager Managed Rotation pour faire pivoter les secrets du cluster dans AWS PCS. La rotation régulière des secrets est une bonne pratique de sécurité pour maintenir une posture de sécurité solide dans les environnements HPC. Cette fonctionnalité vous permet de respecter les normes de conformité du secteur, notamment les normes HIPAA et FedRAMP, qui imposent une rotation régulière des accréditations.

Le secret du cluster a un double objectif : authentifier les nœuds de calcul rejoignant le cluster et servir de clé JWT pour l'authentification de l'API REST de Slurm. En cas de rotation, les deux aspects sont affectés simultanément.

Comment fonctionne la rotation des secrets du cluster

Préparez-vous manuellement à maintenir la stabilité du cluster pendant la rotation secrète :

1. Préparation : redimensionnez tous les groupes de nœuds de calcul à une capacité nulle et assurez-vous qu'aucune tâche n'est en cours d'exécution
2. Rotation : initiez la rotation via la console ou l'API Secrets Manager
3. Surveillance — Suivez les progrès grâce aux CloudTrail événements
4. Restauration : redimensionnez les groupes de nœuds de calcul pour les ramener à la capacité souhaitée

Pendant la rotation, votre cluster reste en bon `ACTIVE` état et la facturation se poursuit normalement. Le processus prend généralement quelques minutes.

Exigences et limitations

Avant de faire pivoter les secrets du cluster, répondez aux exigences suivantes :

- Le cluster doit être dans `ACTIVE` ou dans un `UPDATE_FAILED` état
- Le rôle IAM doit avoir une autorisation `secretsmanager:RotateSecret`
- Tous les groupes de nœuds de calcul doivent être dimensionnés à une capacité nulle
- Arrêter toutes les tâches avant la rotation

Limites:

- Préparation manuelle requise pour chaque rotation
- Les jetons JWT existants deviennent invalides et doivent être réémis
- Les nœuds de connexion BYO nécessitent une mise à jour secrète manuelle après la rotation

Rubriques

- Faire pivoter un secret de cluster dans AWS PCS
- Questions fréquemment posées sur la rotation des secrets de cluster dans AWS PCS
- Résolution des problèmes de rotation des secrets de cluster dans AWS PCS

Faire pivoter un secret de cluster dans AWS PCS

Faites alterner le secret de votre cluster pour vous conformer aux exigences de sécurité et faire face aux compromissions potentielles. Ce processus nécessite de mettre votre cluster en mode maintenance.

Prérequis

- Rôle IAM avec autorisation `secretsmanager:RotateSecret`
- Cluster dans `ACTIVE` notre `UPDATE_FAILED` état

Procédure

1. Informez les utilisateurs du cluster de la prochaine fenêtre de maintenance.
2. Mettez le cluster en mode maintenance en redimensionnant tous les groupes de nœuds de calcul à une capacité nulle.
   a. Utilisez l' UpdateComputeNodeGroup API pour définir les deux minInstanceCount et la valeur 0 maxInstanceCount pour tous les groupes de nœuds de calcul.
   b. Attendez que tous les nœuds s'arrêtent.
   c. Facultatif : Videz les files d'attente du planificateur à l'aide des commandes Slurm avant de mettre fin à la capacité afin de gérer les tâches avec élégance.
3. Lancez la rotation via Secrets Manager.
   - Méthode de console :
     - Accédez à Secrets Manager, sélectionnez le secret de votre cluster, puis choisissez Rotate secret.
   - Méthode API :
     - Utilisez l'`rotate-secret`API Secrets Manager.
4. Surveillez la progression de la rotation.
   a. Suivez les progrès par le biais CloudTrail d'événements.
   b. Vérifiez `lastRotatedDate` via la console Secrets Manager ou l'`secretsmanager:describeSecret`API.
   c. Attendez notre `RotationSucceeded` `RotationFailed` CloudTrail événement.
5. Une fois la rotation réussie, restaurez la capacité du cluster.
   a. Utilisez l' UpdateComputeNodeGroup API pour réinitialiser les groupes de nœuds à la min/max capacité souhaitée.
   b. Pour les nœuds de AWS connexion gérés par PC : aucune action supplémentaire n'est requise.
   c. Pour les nœuds de connexion BYO :
      i. Connectez-vous aux nœuds de connexion.
      ii. Mise à jour `/etc/slurm/slurm.key` avec le nouveau secret de Secrets Manager.
      iii. Redémarrez le démon Slurm Auth et Cred Kiosk (sackd).

Questions fréquemment posées sur la rotation des secrets de cluster dans AWS PCS

Trouvez les réponses aux questions les plus fréquemment posées sur la rotation des secrets de cluster dans AWS PCS.

Qu'est-ce qu'un secret de cluster ?

Un secret de cluster est un identifiant sécurisé qui permet des communications sécurisées entre le contrôleur Slurm et les nœuds de calcul AWS PCS. Il sert également de clé JSON Web Token (JWT) pour l'authentification de l'API REST de Slurm.

Quelle est la différence entre le secret du cluster et la clé JWT ?

Dans AWS PCS, le secret du cluster et la clé JWT sont la même ressource servant des objectifs différents. Le secret du cluster authentifie les communications internes de Slurm, tandis que la clé JWT signe les jetons pour l'authentification via l'API REST. En cas de rotation, les deux aspects sont affectés simultanément.

Combien de temps dure la rotation ?

Le processus de rotation prend généralement quelques minutes. Votre cluster reste à l'état ACTIF et la facturation se poursuit normalement pendant la rotation.

Puis-je planifier des rotations automatiques ?

Vous pouvez activer la rotation planifiée dans Secrets Manager. Cependant, la version initiale nécessite une préparation manuelle (mise à l'échelle des groupes de nœuds à 0) avant chaque rotation.

Mes jetons JWT existants fonctionneront-ils toujours après la rotation ?

Non, les jetons JWT existants deviennent invalides après la rotation. Émettez de nouveaux jetons pour les clients de l'API REST.

Où puis-je trouver le secret de mon cluster ?

Vous pouvez trouver le secret de votre cluster dans la console Secrets Manager ou via la console AWS PCS. Pour des instructions détaillées, reportez-vous AWS Secrets Manager À utiliser pour trouver le secret du cluster aux sections etUtilisez AWS PCS pour trouver le secret du cluster.

Pourquoi la rotation nécessite-t-elle de redimensionner les groupes de nœuds à 0 ?

La rotation ne nécessite aucune instance en cours d'exécution pour garantir la stabilité du cluster pendant le processus de mise à jour secrète. Cela permet d'éviter les conflits d'authentification entre les anciens et les nouveaux secrets.

Quelles sont les exigences de conformité prises en charge par cette fonctionnalité ?

Cette fonctionnalité permet à AWS PCS de respecter les normes de conformité du secteur, notamment HIPAA et FedRAMP, qui imposent une rotation régulière des identifiants dans le cadre de leurs contrôles de sécurité.

Résolution des problèmes de rotation des secrets de cluster dans AWS PCS

La rotation du secret du cluster échoue si l'environnement n'est pas correctement préparé. Les instances actives de votre cluster en sont la cause la plus courante. Pour éviter toute défaillance :

1. Définissez la capacité de tous les groupes de nœuds sur 0.
2. Attendez que les nœuds s'arrêtent.
3. Vérifiez que votre cluster n'est pas dans les états suivants : `CREATE_FAILED`DELETE_FAILED,`RESUMING`,`SUSPENDING`, ou`SUSPENDED`.

En cas d'échec de la rotation :

- Un RotationFailed CloudTrail événement apparaît
- Le secret du cluster reste inchangé
- Consultez l' RotationFailed événement CloudTrail pour plus de détails
- Effectuez toutes les étapes de préparation pour une rotation réussie

AWS Groupes de nœuds de calcul PCS

Un groupe de nœuds de calcul AWS PCS est un ensemble logique de nœuds (EC2 instances Amazon). Ces nœuds peuvent être utilisés pour exécuter des tâches informatiques, ainsi que pour fournir un accès interactif basé sur un shell à un système HPC. Un groupe de nœuds de calcul comprend des règles pour créer des nœuds, notamment les types d' EC2 instances Amazon à utiliser, le nombre d'instances à exécuter, l'utilisation d'instances ponctuelles ou d'instances à la demande, les sous-réseaux et les groupes de sécurité à utiliser, et la manière de configurer chaque instance lors de son lancement. Lorsque ces règles sont mises à jour, AWS PCS met à jour les ressources associées au groupe de nœuds de calcul pour qu'elles correspondent.

Rubriques

- Création d'un groupe de nœuds de calcul dans AWS PCS
- Mise à jour d'un groupe de nœuds de calcul AWS PCS
- Suppression d'un groupe de nœuds de calcul dans AWS PCS
- Obtenez les détails du groupe de nœuds de calcul dans AWS PCS
- Recherche d'instances de groupes de nœuds de calcul dans AWS PCS

Création d'un groupe de nœuds de calcul dans AWS PCS

Cette rubrique fournit une vue d'ensemble des options disponibles et décrit les éléments à prendre en compte lors de la création d'un groupe de nœuds de calcul dans AWS Parallel Computing Service (AWS PCS). Si c'est la première fois que vous créez un groupe de nœuds de calcul dans AWS PCS, nous vous recommandons de suivre le didacticiel dansCommencez avec AWS Parallel Computing Service. Le didacticiel peut vous aider à créer un système HPC fonctionnel sans étendre toutes les options disponibles et les architectures système possibles.

> **Note**
>
> Vous pouvez configurer des paramètres Slurm personnalisés sur des groupes de nœuds de calcul afin de contrôler l'utilisation des ressources et les comportements au niveau des nœuds. Pour de plus amples informations, veuillez consulter Configuration des paramètres personnalisés de Slurm dans PCS AWS.

> **Important**
>
> AWS Le PCS nécessite actuellement un noyau prenant en IPv4 charge la communication entre nœuds locaux, même lorsque vous utilisez le AWS PCS dans un réseau IPv6 uniquement. Pour de plus amples informations, veuillez consulter Images Amazon Machine personnalisées (AMIs) pour AWS PC.

Conditions préalables

- Quotas de service suffisants pour lancer le nombre souhaité d'instances EC2 dans votre Région AWS. Vous pouvez utiliser le AWS Management Consolepour vérifier et demander des augmentations de vos quotas de service.
- Un VPC et un ou plusieurs sous-réseaux existants répondant aux exigences du réseau AWS PCS. Nous vous recommandons de bien comprendre ces exigences avant de déployer un cluster à des fins de production. Pour de plus amples informations, veuillez consulter AWS Exigences et considérations relatives au PCS, au VPC et aux sous-réseaux. Vous pouvez également utiliser un CloudFormation modèle pour créer un VPC et des sous-réseaux. AWS fournit une recette HPC pour le CloudFormation modèle. Pour plus d'informations, voir aws-hpc-recipesci-dessous GitHub.
- Un profil d'instance IAM autorisé à appeler l'action de l'`RegisterComputeNodeGroupInstance`API AWS PCS et à accéder à toutes les autres AWS ressources requises pour les instances de votre groupe de nœuds. Pour de plus amples informations, veuillez consulter Profils d'instance IAM pour AWS Parallel Computing Service.
- Un modèle de lancement pour les instances de votre groupe de nœuds. Pour de plus amples informations, veuillez consulter Utilisation des modèles de lancement Amazon EC2 avec PCS AWS.
- Pour créer un groupe de nœuds de calcul utilisant des instances Spot Amazon EC2, vous devez avoir le rôle lié au service AWSServiceRoleForEC2Spot dans votre. Compte AWS Pour de plus amples informations, veuillez consulter Rôle Amazon EC2 Spot pour PC AWS.

Création d'un groupe de nœuds de calcul dans AWS PCS

Vous pouvez créer un groupe de nœuds de calcul à l'aide du AWS Management Console ou du AWS CLI.

AWS Management Console

Pour créer votre groupe de nœuds de calcul à l'aide de la console

1. Ouvrez la console AWS PCS.
2. Sélectionnez le cluster dans lequel vous souhaitez créer un groupe de nœuds de calcul. Accédez à Compute node groups et choisissez Create.
3. Dans la section Configuration du groupe de nœuds de calcul, donnez un nom à votre groupe de nœuds. Le nom ne peut contenir que des caractères alphanumériques et des tirets distinguant majuscules et minuscules. Il doit commencer par un caractère alphabétique et ne doit pas dépasser 25 caractères. Le nom doit être unique au sein du cluster.
4. Sous Configuration informatique, entrez ou sélectionnez les valeurs suivantes :

   a. Modèle de lancement EC2 : sélectionnez un modèle de lancement personnalisé à utiliser pour ce groupe de nœuds. Les modèles de lancement peuvent être utilisés pour personnaliser les paramètres réseau tels que les sous-réseaux et les groupes de sécurité, la configuration de surveillance et le stockage au niveau de l'instance. Si vous n'avez pas préparé de modèle de lancement, consultez la section Utilisation des modèles de lancement Amazon EC2 avec PCS AWS pour savoir comment en créer un.

   > **Important**
   >
   > AWS PCS crée un modèle de lancement géré pour chaque groupe de nœuds de calcul. Ils sont nommés`pcs-`*`identifier`*`-do-not-delete`. Ne les sélectionnez pas lorsque vous créez ou mettez à jour un groupe de nœuds de calcul, sinon le groupe de nœuds ne fonctionnera pas correctement.

   b. Version du modèle de lancement EC2 : vous devez sélectionner une version de votre modèle de lancement personnalisé. Si vous modifiez la version ultérieurement, vous devez mettre à jour le groupe de nœuds de calcul pour détecter les modifications apportées au modèle de lancement. Pour de plus amples informations, veuillez consulter Mise à jour d'un groupe de nœuds de calcul AWS PCS.

   c. ID AMI : si votre modèle de lancement n'inclut pas d'ID AMI, ou si vous souhaitez remplacer la valeur du modèle de lancement, fournissez un ID d'AMI ici. Notez que l'AMI utilisée pour le groupe de nœuds doit être compatible avec le AWS PCS. Vous pouvez également sélectionner un exemple d'AMI fourni par AWS. Pour plus d'informations sur ce sujet, consultezAmazon Machine Images (AMIs) pour AWS PC.

d. Profil d'instance IAM : choisissez un profil d'instance pour le groupe de nœuds. Un profil d'instance accorde à l'instance les autorisations nécessaires pour accéder aux AWS ressources et aux services en toute sécurité. Si vous n'en avez pas préparé un, vous pouvez sélectionner Créer un profil de base pour que AWS PCS en crée un pour vous avec la politique minimale, ou voirProfils d'instance IAM pour AWS Parallel Computing Service.

e. Sous-réseaux — Choisissez un ou plusieurs sous-réseaux dans le VPC sur lequel votre cluster AWS PCS est déployé. Si vous sélectionnez plusieurs sous-réseaux, les communications EFA ne seront pas disponibles entre les nœuds et les communications entre les nœuds de différents sous-réseaux peuvent augmenter le temps de latence. Assurez-vous que les sous-réseaux que vous spécifiez ici correspondent à ceux que vous définissez dans le modèle de lancement EC2.

f. Instances — Choisissez un ou plusieurs types d'instances pour répondre aux demandes de dimensionnement dans le groupe de nœuds. Tous les types d'instances doivent avoir la même architecture de processeur (x86_64 ou arm64) et le même numéro de v. CPUs Si les instances en ont GPUs, tous les types d'instances doivent avoir le même nombre de GPUs.

g. Configuration de dimensionnement : spécifiez le nombre minimum et maximum d'instances pour le groupe de nœuds. Vous pouvez définir soit une configuration statique, dans laquelle un nombre fixe de nœuds sont en cours d'exécution, soit une configuration dynamique, dans laquelle le nombre maximum de nœuds peut être exécuté. Pour une configuration statique, définissez le minimum et le maximum sur le même nombre, supérieur à zéro. Pour une configuration dynamique, définissez le nombre minimum d'instances à zéro et le nombre maximal d'instances à un nombre supérieur à zéro. AWS Le PCS ne prend pas en charge les groupes de nœuds de calcul composés d'une combinaison d'instances statiques et dynamiques.

5. (Facultatif) Sous Paramètres supplémentaires, spécifiez les éléments suivants :

a. Option d'achat : sélectionnez des instances à la demande, des instances ponctuelles ou un bloc de capacité existant. Choisissez également On-Demand si vous prévoyez d'utiliser une réservation de capacité à la demande (ODCR). Pour de plus amples informations, veuillez consulter Utilisation ODCRs avec AWS PCS. Choisissez Capacity Block pour utiliser un bloc de capacité Amazon EC2 existant pour la réservation de ML. Pour de plus amples informations, veuillez consulter Utilisation des blocs de capacité Amazon EC2 pour le ML avec PCS AWS.

b. Stratégie d'allocation : si vous avez sélectionné l'option d'achat ponctuel, vous pouvez spécifier comment les pools de capacité ponctuels sont choisis lors du lancement des instances dans le groupe de nœuds. Pour plus d'informations, consultez la section Stratégies d'allocation pour les instances Spot dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud. Cette option n'a aucun effet si vous avez sélectionné l'option d'achat à la demande.

6. (Facultatif) Dans la section Slurmdes paramètres personnalisés, vous pouvez ajouter des paires de nom et de valeur des paramètres pour configurer des paramètres Slurm supplémentaires. Pour obtenir la liste complète des paramètres pris en charge, consultezParamètres Slurm personnalisés pour les groupes de nœuds de calcul AWS PCS.
7. (Facultatif) Sous Balises, ajoutez des balises à votre groupe de nœuds de calcul.
8. Choisissez Créer un groupe de nœuds de calcul. Le champ Status s'affiche `Creating` lorsque AWS PCS approvisionne le groupe de nœuds. Cela peut prendre plusieurs minutes.

Étape suivante recommandée

- Ajoutez votre groupe de nœuds à une file d'attente dans AWS PCS pour lui permettre de traiter les tâches.

AWS CLI

Pour créer votre groupe de nœuds de calcul à l'aide de AWS CLI

Créez votre file d'attente avec la commande suivante. Avant d'exécuter la commande, effectuez les remplacements suivants :

1. *`region`*Remplacez-le par l'ID du dans lequel Région AWS créer votre cluster, tel `queus-east-1`.
2. Remplacez *`my-cluster`* par le nom ou `clusterId` celui de votre cluster.
3. *`my-node-group`*Remplacez-le par le nom de votre groupe de nœuds de calcul. Un nom ne peut contenir que des caractères alphanumériques (sensibles à la casse) et des traits d'union. Il doit commencer par un caractère alphabétique et ne doit pas dépasser 25 caractères. Le nom doit être unique au sein du cluster.
4. *`subnet-ExampleID1`*Remplacez-le par un ou plusieurs sous-réseaux IDs provenant de votre VPC de cluster.

5. *lt-ExampleID1*Remplacez-le par l'ID de votre modèle de lancement personnalisé. Si vous n'en avez pas préparé un, consultez Utilisation des modèles de lancement Amazon EC2 avec PCS AWS la section pour savoir comment en créer un.

   > **Important**
   >
   > AWS PCS crée un modèle de lancement géré pour chaque groupe de nœuds de calcul. Ils sont nomméspcs-*identifier*-do-not-delete. Ne les sélectionnez pas lorsque vous créez ou mettez à jour un groupe de nœuds de calcul, sinon le groupe de nœuds ne fonctionnera pas correctement.

6. *launch-template-version*Remplacez-le par une version de modèle de lancement spécifique. AWS PCS associe votre groupe de nœuds à cette version spécifique du modèle de lancement.
7. *arn:InstanceProfile*Remplacez-le par l'ARN de votre profil d'instance IAM. Si vous n'en avez pas préparé un, consultez Utilisation des modèles de lancement Amazon EC2 avec PCS AWS pour obtenir des conseils.
8. Remplacez *min-instances* et *max-instances* par des valeurs entières. Vous pouvez définir soit une configuration statique, dans laquelle un nombre fixe de nœuds sont en cours d'exécution, soit une configuration dynamique, dans laquelle le nombre maximum de nœuds peut être exécuté. Pour une configuration statique, définissez le minimum et le maximum sur le même nombre, supérieur à zéro. Pour une configuration dynamique, définissez le nombre minimum d'instances à zéro et le nombre maximal d'instances à un nombre supérieur à zéro. AWS Le PCS ne prend pas en charge les groupes de nœuds de calcul composés d'une combinaison d'instances statiques et dynamiques.
9. Remplacez *t3.large* par un autre type d'instance. Vous pouvez ajouter d'autres types d'instances en spécifiant une liste de `instanceType` paramètres. Par exemple, *--instance-configs instanceType=c6i.16xlarge instanceType=c6a.16xlarge*. Tous les types d'instances doivent avoir la même architecture de processeur (x86_64 ou arm64) et le même numéro de v. CPUs Si les instances en ont GPUs, tous les types d'instances doivent avoir le même nombre de GPUs.

```
aws pcs create-compute-node-group --region region \
    --cluster-identifier my-cluster \
    --compute-node-group-name my-node-group \
    --subnet-ids subnet-ExampleID1 \
```

```
    --custom-launch-template id=lt-ExampleID1,version='launch-template-version' \
    --iam-instance-profile-arn=arn:InstanceProfile \
    --scaling-config minInstanceCount=min-instances,maxInstanceCount=max-instance \
    --instance-configs instanceType=t3.large
```

Example— Création d'un groupe de nœuds de calcul avec des paramètres Slurm personnalisés

```
aws pcs create-compute-node-group --region region \
    --cluster-identifier my-cluster \
    --compute-node-group-name my-node-group \
    --subnet-ids subnet-ExampleID1 \
    --custom-launch-template id=lt-ExampleID1,version='launch-template-version' \
    --iam-instance-profile-arn=arn:InstanceProfile \
    --scaling-config minInstanceCount=min-instances,maxInstanceCount=max-instance \
    --instance-configs instanceType=t3.large \
    --slurm-configuration \
    'slurmCustomSettings=[{parameterName=Features,parameterValue="gpu,nvme"}]'
```

Pour de plus amples informations, veuillez consulter Paramètres Slurm personnalisés pour les groupes de nœuds de calcul AWS PCS.

Vous pouvez ajouter plusieurs paramètres de configuration facultatifs à la `create-compute-node-group` commande.

- Vous pouvez spécifier `--amiId` si votre modèle de lancement personnalisé n'inclut pas de référence à une AMI ou si vous souhaitez remplacer cette valeur. Notez que l'AMI utilisée pour le groupe de nœuds doit être compatible avec le AWS PCS. Vous pouvez également sélectionner un exemple d'AMI fourni par AWS. Pour plus d'informations sur ce sujet, consultezAmazon Machine Images (AMIs) pour AWS PC.
- `--purchase-option`À utiliser pour choisir la manière dont AWS PCS achète les instances EC2 pour votre groupe de nœuds de calcul. On-Demand est la valeur par défaut.
  - `ONDEMAND`— Utilisez des instances à la demande. Choisissez également cette option si vous prévoyez d'utiliser une réservation de capacité à la demande (ODCR). Pour de plus amples informations, veuillez consulter Utilisation ODCRs avec AWS PCS.
  - `SPOT`— Utilisez des instances ponctuelles. Si vous choisissez des instances Spot, vous pouvez également les utiliser `--allocation-strategy` pour définir la manière dont AWS PCS choisit les pools de capacité Spot lorsqu'il lance des instances dans le groupe de nœuds. Pour plus d'informations, consultez la section Stratégies d'allocation pour les instances Spot dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

  - `CAPACITY_BLOCK`— Utilisez un bloc de capacité Amazon EC2 existant pour la réservation de machine learning. Pour de plus amples informations, veuillez consulter Utilisation des blocs de capacité Amazon EC2 pour le ML avec PCS AWS.
- Il est possible de fournir des options Slurm de configuration pour les nœuds du groupe de nœuds à l'aide de`--slurm-configuration`. Vous pouvez définir le poids (priorité de planification) et la mémoire réelle. Les nœuds dont le poids est faible ont une priorité plus élevée et les unités sont arbitraires. Pour plus d'informations, consultez la section Poids dans la Slurm documentation. La mémoire réelle définit la taille (en Go) de la mémoire réelle sur les nœuds du groupe de nœuds. Il est destiné à être utilisé conjointement avec l'`CR_CPU_Memory`option pour le cluster dans AWS PCS dans votre Slurm configuration. Pour plus d'informations, consultez la section RealMemorydans la documentation Slurm.

> **Important**
>
> La création du groupe de nœuds de calcul peut prendre plusieurs minutes.

Vous pouvez demander l'état de votre groupe de nœuds à l'aide de la commande suivante. Vous ne pourrez pas associer le groupe de nœuds à une file d'attente tant que son statut ne sera pas atteint`ACTIVE`.

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

Mise à jour d'un groupe de nœuds de calcul AWS PCS

Cette rubrique fournit une vue d'ensemble des options disponibles et décrit les éléments à prendre en compte lors de la mise à jour d'un groupe de nœuds de calcul AWS PCS. Pour plus d'informations sur les paramètres personnalisés de Slurm, consultez. Paramètres Slurm personnalisés pour les groupes de nœuds de calcul AWS PCS

Options de mise à jour d'un groupe de nœuds de calcul AWS PCS

La mise à jour d'un groupe de nœuds de calcul AWS PCS vous permet de modifier les propriétés des instances lancées par AWS PCS, ainsi que les règles régissant le lancement de ces instances.

Par exemple, vous pouvez remplacer l'AMI pour les instances de groupes de nœuds par une autre sur laquelle un logiciel différent est installé. Vous pouvez également mettre à jour les groupes de sécurité pour modifier la connectivité réseau entrante ou sortante. Vous pouvez également modifier la configuration de dimensionnement et l'option d'achat préférée.

Les paramètres des groupes de nœuds suivants ne peuvent pas être modifiés après leur création :

- Nom
- instances

Considérations relatives à la mise à jour d'un groupe de nœuds de calcul AWS PCS

Les groupes de nœuds de calcul définissent les instances EC2 utilisées pour traiter les tâches, fournir un accès au shell interactif et effectuer d'autres tâches. Ils sont souvent associés à une ou plusieurs files d'attente AWS PCS. Lorsque vous mettez à jour votre groupe de nœuds de calcul pour modifier son comportement (ou celui de ses nœuds), tenez compte des points suivants :

- Les modifications apportées aux propriétés du groupe de nœuds de calcul entrent en vigueur lorsque le statut du groupe de nœuds de calcul passe de Mise à jour à Actif. Les nouvelles instances sont lancées avec les propriétés mises à jour.
- Les mises à jour qui n'ont aucun impact sur la configuration de nœuds spécifiques n'affectent pas les nœuds en cours d'exécution. Par exemple, ajouter un sous-réseau et modifier la stratégie d'allocation.
- Si vous mettez à jour le modèle de lancement d'un groupe de nœuds de calcul, vous devez mettre à jour le groupe de nœuds de calcul pour utiliser la nouvelle version.
- Pour ajouter ou supprimer un groupe de sécurité dans les nœuds d'un groupe de nœuds de calcul, modifiez son modèle de lancement et mettez à jour le groupe de nœuds de calcul. Les nouvelles instances sont lancées avec l'ensemble de groupes de sécurité mis à jour.
- Si vous modifiez directement un groupe de sécurité utilisé par un groupe de nœuds de calcul, cela a un effet immédiat sur les instances en cours d'exécution et les instances futures.
- Si vous ajoutez ou supprimez des autorisations dans le profil d'instance IAM utilisé par un groupe de nœuds de calcul, cela a un effet immédiat sur les instances en cours d'exécution et les instances futures.

- Pour modifier l'AMI utilisée par les instances d'un groupe de nœuds de calcul, mettez à jour le groupe de nœuds de calcul (ou son modèle de lancement) pour utiliser la nouvelle AMI et attendez que AWS PCS remplace les instances.
- AWS Le PCS remplace les instances existantes dans le groupe de nœuds après une opération de mise à jour du groupe de nœuds. Si des tâches sont exécutées sur un nœud, elles sont autorisées à se terminer avant que AWS PCS ne remplace le nœud. Les processus utilisateur interactifs (tels que sur les instances du nœud de connexion) sont interrompus. L'état du groupe de nœuds revient au `Active` moment où AWS PCS marque les instances à remplacer, mais le remplacement réel se produit lorsque les instances sont inactives.
- Si vous diminuez le nombre maximum d'instances autorisées dans un groupe de nœuds de calcul, AWS PCS supprime les nœuds de Slurm pour atteindre le nouveau maximum. AWS PCS met fin à l'exécution des instances associées aux nœuds Slurm supprimés. Les tâches en cours sur les nœuds supprimés échouent et retournent dans leurs files d'attente.
- AWS PCS crée un modèle de lancement géré pour chaque groupe de nœuds de calcul. Ils sont nommés`pcs-`*`identifier`*`-do-not-delete`. Ne les sélectionnez pas lorsque vous créez ou mettez à jour un groupe de nœuds de calcul, sinon le groupe de nœuds ne fonctionnera pas correctement.
- Si vous mettez à jour un groupe de nœuds de calcul pour utiliser Spot pour son option d'achat, vous devez avoir le rôle lié au service AWSServiceRoleForEC2Spot dans votre compte. Pour de plus amples informations, veuillez consulter Rôle Amazon EC2 Spot pour PC AWS.

Pour mettre à jour un groupe de nœuds de calcul AWS PCS

Vous pouvez mettre à jour un groupe de nœuds à l'aide de l'AWS Management Console ou de l'AWS CLI.

AWS Management Console

Pour mettre à jour un groupe de nœuds de calcul

1. Ouvrez la console AWS PCS à l'adresse `https://console.aws.amazon.com/pcs/home#/clusters`
2. Sélectionnez le cluster dans lequel vous souhaitez mettre à jour un groupe de nœuds de calcul.
3. Accédez à Calculer les groupes de nœuds, accédez au groupe de nœuds que vous souhaitez mettre à jour, puis sélectionnez Modifier.

4. Dans les sections Configuration informatique, Paramètres supplémentaires et Paramètres Slurmde personnalisation, mettez à jour toutes les valeurs sauf :

   - Instances : vous ne pouvez pas modifier les instances d'un groupe de nœuds de calcul.

   Pour plus d'informations sur les paramètres personnalisés de Slurm, consultez. Paramètres Slurm personnalisés pour les groupes de nœuds de calcul AWS PCS
5. Choisissez Mettre à jour. Le champ État affichera la mise à jour pendant que les modifications sont appliquées.

> **Important**
>
> Les mises à jour des groupes de nœuds de calcul peuvent prendre plusieurs minutes.

AWS CLI

Pour mettre à jour un groupe de nœuds de calcul

1. Mettez à jour votre groupe de nœuds de calcul à l'aide de la commande suivante. Avant d'exécuter la commande, effectuez les remplacements suivants :

   a. *region-code*Remplacez-le par la région AWS dans laquelle vous souhaitez créer votre cluster.

   b. *my-node-group*Remplacez-le par le nom ou par le nom `computeNodeGroupId` de votre groupe de nœuds de calcul.

   c. *my-cluster*Remplacez-le par le nom ou `clusterId` celui de votre cluster.

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

Example— Mise à jour d'un groupe de nœuds de calcul avec des paramètres Slurm personnalisés

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

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

Pour de plus amples informations, veuillez consulter Paramètres Slurm personnalisés pour les groupes de nœuds de calcul AWS PCS.

2. Mettez à jour les paramètres de tous les groupes de nœuds, à l'exception de`--instance-configs`. Par exemple, pour définir un nouvel ID d'AMI, indiquez `--amiId my-custom-ami-id` où il *my-custom-ami-id* est remplacé par l'AMI de votre choix.

> **Important**
>
> La mise à jour du groupe de nœuds de calcul peut prendre plusieurs minutes.

Vous pouvez demander l'état de votre groupe de nœuds à l'aide de la commande suivante.

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

Suppression d'un groupe de nœuds de calcul dans AWS PCS

Cette rubrique fournit une vue d'ensemble des options disponibles et décrit les éléments à prendre en compte lorsque vous supprimez un groupe de nœuds de calcul dans AWS PCS.

Considérations relatives à la suppression d'un groupe de nœuds de calcul

Les groupes de nœuds de calcul définissent les instances EC2 utilisées pour traiter les tâches, fournir un accès au shell interactif et effectuer d'autres tâches. Ils sont souvent associés à une ou plusieurs files d'attente AWS PCS. Avant de supprimer un groupe de nœuds de calcul, tenez compte des points suivants :

- Toutes les instances EC2 lancées par le groupe de nœuds de calcul seront résiliées. Cela annulera les tâches en cours d'exécution sur ces instances et mettra fin à l'exécution des processus interactifs.

- Vous devez dissocier le groupe de nœuds de calcul de toutes les files d'attente avant de pouvoir le supprimer. Pour de plus amples informations, veuillez consulter Mettre à jour une file d'attente AWS PCS.

Supprimer le groupe de nœuds de calcul

Vous pouvez utiliser le AWS Management Console ou AWS CLI pour supprimer un groupe de nœuds de calcul.

AWS Management Console

Pour supprimer un groupe de nœuds de calcul

1. Ouvrez la console AWS PCS.
2. Sélectionnez le cluster du groupe de nœuds de calcul.
3. Accédez à Groupes de nœuds de calcul et sélectionnez le groupe de nœuds de calcul à supprimer.
4. Sélectionnez Delete (Supprimer).
5. Le champ État s'affiche`Deleting`. Cela peut prendre plusieurs minutes.

> **Note**
>
> Vous pouvez utiliser les commandes natives de votre planificateur pour confirmer que le groupe de nœuds de calcul est supprimé. Par exemple, utilisez `sinfo` ou `squeue` pour Slurm.

AWS CLI

Pour supprimer un groupe de nœuds de calcul

- Utilisez la commande suivante pour supprimer un groupe de nœuds de calcul, avec les remplacements suivants :
  - *`region-code`*Remplacez-le par celui dans lequel se trouve Région AWS votre cluster.
  - Remplacez *`my-node-group`* par le nom ou l'ID de votre groupe de nœuds de calcul.
  - Remplacez *`my-cluster`* par le nom ou l'ID de votre cluster.

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

La suppression du groupe de nœuds de calcul peut prendre plusieurs minutes.

> **Note**
>
> Vous pouvez utiliser les commandes natives de votre planificateur pour confirmer que le groupe de nœuds de calcul est supprimé. Par exemple, utilisez `sinfo` ou `squeue` pour Slurm.

Obtenez les détails du groupe de nœuds de calcul dans AWS PCS

Vous pouvez utiliser le AWS Management Console ou AWS CLI pour obtenir des informations sur un groupe de nœuds de calcul, tels que son ID de groupe de nœuds de calcul, le nom de ressource Amazon (ARN) et l'identifiant Amazon Machine Image (AMI). Ces détails sont souvent des valeurs obligatoires pour les actions et les configurations de l'API AWS PCS.

AWS Management Console

Pour obtenir les détails du groupe de nœuds de calcul

1. Ouvrez la console AWS PCS.
2. Sélectionnez le cluster .
3. Choisissez Compute node groups.
4. Choisissez un groupe de nœuds de calcul dans le volet de liste.

AWS CLI

Pour obtenir les détails du groupe de nœuds de calcul

1. Utilisez l'action ListClustersAPI pour trouver le nom ou l'ID de votre cluster.

```
aws pcs list-clusters
```

Exemple de sortie :

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

2. Utilisez l'action ListComputeNodeGroupsAPI pour répertorier les groupes de nœuds de calcul d'un cluster.

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

Exemple d'appel :

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

Exemple de sortie :

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

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

3. Utilisez l'action GetComputeNodeGroupAPI pour obtenir des informations supplémentaires sur un groupe de nœuds de calcul.

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

Exemple d'appel :

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

Exemple de sortie :

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

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

Recherche d'instances de groupes de nœuds de calcul dans AWS PCS

Chaque groupe de nœuds de calcul AWS PCS peut lancer des instances EC2 avec des configurations partagées. Vous pouvez utiliser les balises EC2 pour rechercher des instances dans un groupe de nœuds de calcul dans AWS Management Console ou avec le AWS CLI.

AWS Management Console

Pour trouver les instances de votre groupe de nœuds de calcul

1. Ouvrez la console AWS PCS.
2. Sélectionnez le cluster .
3. Choisissez Compute node groups.
4. Trouvez l'ID du groupe de nœuds de connexion que vous avez créé.
5. Accédez à la console EC2 et choisissez Instances.
6. Recherchez les instances avec la balise suivante. *node-group-id*Remplacez-le par l'ID (et non le nom) de votre groupe de nœuds de calcul.

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

7. (Facultatif) Vous pouvez modifier la valeur de l'état de l'instance dans le champ de recherche pour trouver les instances en cours de configuration ou récemment résiliées.

8. Trouvez l'ID d'instance et l'adresse IP de chaque instance dans la liste des instances balisées.

AWS CLI

Pour rechercher les instances de votre groupe de nœuds, utilisez les commandes ci-dessous. Avant d'exécuter les commandes, effectuez les remplacements suivants :

- *region-code*Remplacez-le par celui Région AWS de votre cluster. Exemple : `us-east-1`
- *node-group-id*Remplacez-le par l'ID (et non le nom) de votre groupe de nœuds de calcul. Pour trouver l'ID d'un groupe de nœuds de calcul, consultezObtenez les détails du groupe de nœuds de calcul dans AWS PCS.
- `running`Remplacez-le par d'autres états d'instance tels que `pending` ou `terminated` pour trouver des instances EC2 dans d'autres états.

```
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}
```

La commande renvoie un résultat semblable à ce qui suit. La valeur de `PublicIP` est `null` si l'instance se trouve dans un sous-réseau privé.

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

> **Note**
>
> Si vous prévoyez `describe-instances` de renvoyer un grand nombre d'instances, vous devez utiliser les options pour plusieurs pages. Pour plus d'informations, consultez DescribeInstances le manuel Amazon Elastic Compute Cloud API Reference.

Utilisation des modèles de lancement Amazon EC2 avec PCS AWS

Dans Amazon EC2, un modèle de lancement peut stocker un ensemble de préférences afin que vous n'ayez pas à les spécifier individuellement lorsque vous lancez des instances. AWS Le PCS intègre des modèles de lancement afin de configurer de manière flexible les groupes de nœuds de calcul. Lorsque vous créez un groupe de nœuds, vous fournissez un modèle de lancement. AWS PCS crée un modèle de lancement dérivé à partir de celui-ci qui inclut des transformations pour garantir son fonctionnement avec le service.

Comprendre les options et les considérations à prendre en compte lors de la rédaction d'un modèle de lancement personnalisé peut vous aider à en créer un à utiliser avec AWS PCS. Pour plus d'informations sur les modèles de lancement, consultez Lancer une instance depuis un modèle de lancement Lancer une instance depuis un modèle de lancement dans le guide de l'utilisateur Amazon EC2.

Rubriques

- Vue d'ensemble des modèles de lancement dans AWS PCS
- Créer un modèle de lancement de base
- Utilisation des données utilisateur Amazon EC2 pour PC AWS
- Réservations de capacité en AWS PCS
- Paramètres utiles du modèle de lancement

Vue d'ensemble des modèles de lancement dans AWS PCS

Il existe plus de 30 paramètres disponibles que vous pouvez inclure dans un modèle de lancement EC2, contrôlant de nombreux aspects de la configuration des instances. La plupart sont entièrement compatibles avec les AWS PC, à quelques exceptions près.

Les paramètres suivants du modèle de lancement EC2 seront ignorés par AWS PCS car ces propriétés doivent être gérées directement par le service :

- Attributs du type d' type/Specify instance (`InstanceRequirements`) — AWS PCS ne prend pas en charge la sélection d'instance basée sur les attributs.

- Type d'instance (`InstanceType`) : spécifiez les types d'instances lorsque vous créez un groupe de nœuds.
- Profil d' details/IAM instance avancé (`IamInstanceProfile`) : vous le fournissez lorsque vous créez ou mettez à jour le groupe de nœuds.
- Résiliation avancée de details/Disable l'API (`DisableApiTermination`) : le AWS PCS doit contrôler le cycle de vie des instances de groupes de nœuds qu'il lance.
- Advanced details/Disable API stop (`DisableApiStop`) : le AWS PCS doit contrôler le cycle de vie des instances de groupes de nœuds qu'il lance.
- Avancé details/Stop — Comportement d'hibernation (`HibernationOptions`) — AWS PCS ne prend pas en charge l'hibernation des instances.
- details/Elastic GPU avancé (`ElasticGpuSpecifications`) — Amazon Elastic Graphics a atteint la fin de son cycle de vie le 8 janvier 2024.
- details/Elastic Inférence avancée (`ElasticInferenceAccelerators`) — Amazon Elastic Inference n'est plus disponible pour les nouveaux clients.
- AAdvanced details/Specify CPU options/Threadsper core (`ThreadsPerCore`) — AWS PCS définit le nombre de threads par cœur à 1.

Ces paramètres ont des exigences particulières en matière de compatibilité avec le AWS PCS :

- Données utilisateur (`UserData`) : elles doivent être codées en plusieurs parties. Consultez Utilisation des données utilisateur Amazon EC2 pour PC AWS.
- Images de l'application et du système d'exploitation (`ImageId`) — Vous pouvez les inclure. Toutefois, si vous spécifiez un ID d'AMI lorsque vous créez ou mettez à jour le groupe de nœuds, il remplacera la valeur du modèle de lancement. L'AMI que vous fournissez doit être compatible avec AWS PCS. Pour plus d'informations, reportez-vous à la section "Amazon Machine Images (AMIs) pour AWS PC.
- Réseau settings/Firewall (groupes de sécurité) (`SecurityGroups`) — Il est impossible de définir une liste de noms de groupes de sécurité dans un modèle de lancement AWS PCS. Vous pouvez définir une liste de groupes de sécurité IDs (`SecurityGroupIds`), sauf si vous définissez des interfaces réseau dans le modèle de lancement. Vous devez ensuite spécifier le groupe de sécurité IDs pour chaque interface. Pour de plus amples informations, veuillez consulter Groupes de sécurité dans AWS PCS.
- Configuration settings/Advanced réseau (`NetworkInterfaces`) : si vous utilisez des instances EC2 avec une seule carte réseau et que vous n'avez pas besoin de configuration réseau

spécialisée, AWS PCS peut configurer la mise en réseau des instances pour vous. Pour configurer plusieurs cartes réseau ou pour activer Elastic Fabric Adapter sur vos instances, utilisez`NetworkInterfaces`. Chaque interface réseau doit contenir une liste de groupes de IDs sécurité`Groups`. Pour de plus amples informations, veuillez consulter Plusieurs interfaces réseau dans les AWS PCS.

- Détails avancés/réservation de capacité (`CapacityReservationSpecification`) — Cela peut être défini, mais ne peut pas faire référence à un paramètre spécifique `CapacityReservationId` lorsque vous travaillez avec AWS PCS. Vous pouvez toutefois faire référence à un groupe de réservation de capacité, lorsque ce groupe contient une ou plusieurs réservations de capacité. Pour de plus amples informations, veuillez consulter Réservations de capacité en AWS PCS.

Créer un modèle de lancement de base

Vous pouvez créer un modèle de lancement à l'aide du AWS Management Console ou du AWS CLI.

AWS Management Console

Pour créer un modèle de lancement

1. Ouvrez la EC2console Amazon et sélectionnez Launch templates.
2. Choisissez Créer un modèle de lancement.
3. Sous Nom et description du modèle de lancement, entrez un nom unique et distinctif pour le nom du modèle de lancement
4. Sous Paire de clés (connexion) sous Nom de la paire de clés, sélectionnez la paire de clés SSH qui sera utilisée pour se connecter aux EC2 instances gérées par AWS PCS. Cette action est facultative, mais recommandée.
5. Sous Paramètres réseau, puis Pare-feu (groupes de sécurité), choisissez les groupes de sécurité à associer à l'interface réseau. Tous les groupes de sécurité du modèle de lancement doivent provenir du VPC de votre cluster AWS PCS. Choisissez au minimum :
   - Un groupe de sécurité qui permet la communication avec le cluster AWS PCS
   - Un groupe de sécurité qui permet la communication entre les EC2 instances lancées par AWS PCS
   - (Facultatif) Un groupe de sécurité qui autorise l'accès SSH entrant aux instances interactives

- (Facultatif) Un groupe de sécurité qui permet aux nœuds de calcul d'établir des connexions sortantes vers Internet
- (Facultatif) Groupe (s) de sécurité qui autorisent l'accès à des ressources en réseau telles que des systèmes de fichiers partagés ou un serveur de base de données.

6. Votre nouvel identifiant de modèle de lancement sera accessible dans la EC2 console Amazon sous Modèles de lancement. L'ID du modèle de lancement contiendra le formulaire`lt-0123456789abcdef01`.

Étape suivante recommandée

- Utilisez le nouveau modèle de lancement pour créer ou mettre à jour un groupe de nœuds de calcul AWS PCS.

AWS CLI

Pour créer un modèle de lancement

Créez votre modèle de lancement à l'aide de la commande ci-dessous.

- Avant d'exécuter la commande, effectuez les remplacements suivants :
  a. Remplacez *`region-code`* par l' Région AWS endroit où vous travaillez avec AWS PCS
  b. *`my-launch-template-name`*Remplacez-le par un nom pour votre modèle. Il doit être unique au Compte AWS et Région AWS que vous utilisez.
  c. Remplacez *`my-ssh-key-name`* par le nom de votre clé SSH préférée.
  d. Remplacez *`sg-ExampleID1`* et *`sg-ExampleID2`* par un groupe de sécurité IDs qui autorise la communication entre vos EC2 instances et le planificateur ainsi que la communication entre les EC2 instances. Si vous ne disposez que d'un seul groupe de sécurité qui autorise tout ce trafic, vous pouvez supprimer `sg-ExampleID2` la virgule qui la précède. Vous pouvez également ajouter d'autres groupes de sécurité IDs. Tous les groupes de sécurité que vous incluez dans le modèle de lancement doivent provenir du AWS VPC de votre cluster PCS.

```
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"]}'
```

Le texte affiché AWS CLI ressemblera à ce qui suit. L'ID du modèle de lancement se trouve dans`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"
    }
}
```

Étape suivante recommandée

- Utilisez le nouveau modèle de lancement pour créer ou mettre à jour un groupe de nœuds de calcul AWS PCS.

Utilisation des données utilisateur Amazon EC2 pour PC AWS

Vous pouvez fournir des données utilisateur EC2 dans votre modèle de lancement qui `cloud-init` s'exécute lors du lancement de vos instances. Les blocs de données utilisateur avec le type de contenu `cloud-config` s'exécutent avant que l'instance ne s'enregistre auprès de l'API AWS PCS, tandis que les blocs de données utilisateur avec le type de contenu `text/x-shellscript` s'exécutent une fois l'enregistrement terminé, mais avant le démarrage du démon Slurm. Pour plus d'informations sur les types de contenus, consultez la documentation sur Cloud-Init.

nos données utilisateur peuvent exécuter des scénarios de configuration courants, y compris, mais sans s'y limiter, les suivants :

- Inclure des utilisateurs ou des groupes
- Installation de packages
- Création de partitions et de systèmes de fichiers
- Montage de systèmes de fichiers réseau

Les données utilisateur figurant dans les modèles de lancement doivent être au format d'archive MIME en plusieurs parties. Cela est dû au fait que vos données utilisateur sont fusionnées avec d'autres données utilisateur AWS PCS requises pour configurer les nœuds de votre groupe de nœuds. Vous pouvez combiner plusieurs blocs de données utilisateur dans un seul fichier MIME multi-part.

Un fichier MIME multi-part est constitué des composants suivants :

- Le type de contenu et la déclaration de limite : `Content-Type: multipart/mixed; boundary="==BOUNDARY=="`
- La déclaration de version MIME : `MIME-Version: 1.0`
- Un ou plusieurs blocs de données utilisateur contenant les composants suivants :
  - La limite d'ouverture qui indique le début d'un bloc de données utilisateur :`--==BOUNDARY==`. Vous devez laisser la ligne avant cette limite vide.
  - La déclaration du type de contenu pour le bloc : `Content-Type: text/cloud-config; charset="us-ascii"` ou`Content-Type: text/x-shellscript; charset="us-ascii"`. Vous devez laisser la ligne après la déclaration de type de contenu vide.
  - Le contenu des données utilisateur, tel qu'une liste de commandes ou de `cloud-config` directives du shell.
- La limite de fermeture qui indique la fin du fichier MIME en plusieurs parties :. `--==BOUNDARY==--` Vous devez laisser la ligne avant la limite de fermeture vide.

> **Note**
>
> Si vous ajoutez des données utilisateur à un modèle de lancement dans la console Amazon EC2, vous pouvez les coller sous forme de texte brut. Vous pouvez également le télécharger à partir d'un fichier. Si vous utilisez le AWS CLI ou un AWS SDK, vous devez d'abord encoder les données utilisateur en base64 et envoyer cette chaîne comme valeur du `UserData` paramètre lorsque vous appelez CreateLaunchTemplate, comme indiqué dans ce fichier JSON.

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

```
        "UserData":
 "ewogICAgIkxhdW5jaFRlbXBsYXRlTmFtZSI6ICJpbmNyZWFzZS1jb250YWluZXItdm9sdW..."
    }
}
```

Exemples

- Exemple : installation d'un logiciel à partir d'un référentiel de packages
- Exemple : exécution de scripts à partir d'un compartiment S3
- Exemple : définir des variables d'environnement globales
- Utilisation de systèmes de fichiers réseau avec AWS PCS
- Exemple : utilisation d'un système de fichiers EFS comme répertoire de base partagé

Exemple : installation d'un logiciel pour AWS PC à partir d'un référentiel de packages

Indiquez ce script comme valeur de `"userData"` dans votre modèle de lancement. Pour de plus amples informations, veuillez consulter Utilisation des données utilisateur Amazon EC2 pour PC AWS.

Ce script utilise cloud-config pour installer des packages logiciels sur les instances de groupes de nœuds lors du lancement. Pour plus d'informations, consultez les formats de données utilisateur dans la documentation de cloud-init. Cet exemple installe `curl` et. `llvm`

> **Note**
>
> Vos instances doivent être en mesure de se connecter à leurs référentiels de packages configurés.

```
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==--
```

Exemple : exécution de scripts supplémentaires pour AWS PCS à partir d'un compartiment S3

Indiquez ce script comme valeur de `"userData"` dans votre modèle de lancement. Pour de plus amples informations, veuillez consulter Utilisation des données utilisateur Amazon EC2 pour PC AWS.

Le script de données utilisateur suivant utilise cloud-config pour importer un script depuis un compartiment S3 et l'exécuter sur des instances de groupes de nœuds lors du lancement. Pour plus d'informations, consultez les formats de données utilisateur dans la documentation de cloud-init.

Remplacez les valeurs suivantes par vos propres informations :

- *amzn-s3-demo-bucket*— Le nom d'un compartiment S3 que votre compte peut lire.
- *object-key*— La clé d'objet S3 du script à importer. Cela inclut le nom du script et son emplacement dans la structure de dossiers du bucket. Par exemple, `scripts/script.sh`. Pour plus d'informations, consultez la section Organisation des objets dans la console Amazon S3 à l'aide de dossiers dans le guide de l'utilisateur d'Amazon Simple Storage Service.
- *shell*— Le shell Linux à utiliser pour exécuter le script, tel que`bash`.

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

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

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

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

Le profil d'instance IAM du groupe de nœuds doit avoir accès au bucket. La politique IAM suivante est un exemple pour le bucket dans le script de données utilisateur ci-dessus.

JSON

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

Exemple : définir des variables d'environnement globales pour AWS PCS

Indiquez ce script comme valeur de `"userData"` dans votre modèle de lancement. Pour de plus amples informations, veuillez consulter Utilisation des données utilisateur Amazon EC2 pour PC AWS.

L'exemple suivant permet `/etc/profile.d` de définir des variables globales sur des instances de groupes de nœuds.

```
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==--
```

Exemple : utilisation d'un système de fichiers EFS comme répertoire de base partagé pour AWS PCS

Indiquez ce script comme valeur de `"userData"` dans votre modèle de lancement. Pour de plus amples informations, veuillez consulter Utilisation des données utilisateur Amazon EC2 pour PC AWS.

Cet exemple étend l'exemple de montage EFS Utilisation de systèmes de fichiers réseau avec AWS PCS pour implémenter un répertoire de base partagé. Le contenu de /home est sauvegardé avant le montage du système de fichiers EFS. Le contenu est ensuite rapidement copié sur place sur le stockage partagé une fois le montage terminé.

Remplacez les valeurs suivantes dans ce script par vos propres informations :

- *`/mount-point-directory`*— Le chemin d'une instance sur laquelle vous souhaitez monter le système de fichiers EFS.
- *`filesystem-id`*— L'ID du système de fichiers EFS.

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

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

packages:
  - amazon-efs-utils

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

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

Exemple : activation du SSH sans mot de passe

Vous pouvez vous appuyer sur l'exemple du répertoire de base partagé pour implémenter des connexions SSH entre des instances de cluster à l'aide de clés SSH. Pour chaque utilisateur utilisant le système de fichiers d'accueil partagé, exécutez un script semblable au suivant :

```
#!/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**
>
> Les instances doivent utiliser un groupe de sécurité qui autorise les connexions SSH entre les nœuds du cluster.

Réservations de capacité en AWS PCS

Vous pouvez réserver des EC2 capacités Amazon dans une zone de disponibilité spécifique et pour une durée spécifique à l'aide des réservations de capacité à la demande ou des Amazon EC2 Capacity Blocks for ML afin de vous assurer de disposer de la capacité de calcul nécessaire lorsque vous en avez besoin.

Les réservations de capacité à la demande (ODCRs) vous permettent de réserver de la capacité de calcul pour vos EC2 instances Amazon dans une zone de disponibilité spécifique, quelle que soit la durée. Vous pouvez créer et annuler des réservations à tout moment, sans engagement à long terme ni paiement initial. ODCRs sont idéales lorsque vous avez besoin de réservations de capacité flexibles que vous pouvez modifier en fonction de l'évolution de vos besoins. Pour plus d'informations, consultez la section Réservations de capacité à la demande dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Amazon EC2 Capacity Blocks for ML vous permet de réserver des instances de calcul accéléré basées sur un GPU pour une utilisation future, jusqu'à 8 semaines à l'avance. Vous pouvez réserver

des blocs de 1 à 64 instances pour des durées allant de 1 jour à 6 mois. Les blocs de capacité sont idéaux pour les charges de travail d'apprentissage automatique qui nécessitent un accès garanti à la capacité du GPU à des moments précis. Pour plus d'informations, consultez Capacity Blocks for ML dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Rubriques

- Utilisation ODCRs avec AWS PCS
- Utilisation des blocs de capacité Amazon EC2 pour le ML avec PCS AWS

Utilisation ODCRs avec AWS PCS

Vous pouvez choisir la manière dont AWS PCS consomme vos instances réservées. Si vous créez un ODCR ouvert, toutes les instances correspondantes lancées par AWS PCS ou d'autres processus sur votre compte sont prises en compte dans la réservation. Avec un ODCR ciblé, seules les instances lancées avec l'identifiant de réservation spécifique sont prises en compte dans la réservation. Pour les charges de travail sensibles au facteur temps, les tâches ciblées ODCRs sont plus courantes.

Vous pouvez configurer un groupe de nœuds de calcul AWS PCS pour utiliser un ODCR ciblé en l'ajoutant à un modèle de lancement. Voici les étapes à suivre pour ce faire :

1. Créez une réservation de capacité à la demande (ODCR) ciblée à l'aide du guide de l'utilisateur Amazon EC2 Create a Capacity Reservation.
2. Associez l'ODCR à un modèle de lancement. Il existe deux manières de procéder :

    a. Association ODCR directe : référencez l'ID ODCR directement dans le modèle de lancement. Cette approche permet un contrôle strict de la capacité et ne prend pas en charge le remblayage d'instances (si le groupe de nœuds de calcul demande plus d'instances que ce qui est disponible dans l'ODCR, aucune instance supplémentaire ne sera lancée).

    b. Association de groupes de réservation de capacité : ajoutez l'ODCR à un groupe de réservation de capacité et référencez le groupe dans le modèle de lancement. Cette approche prend en charge le remblayage des instances, ce qui permet à AWS PCS de lancer des instances à la demande supplémentaires si la capacité de réservation est dépassée.

3. Créez ou mettez à jour un groupe de nœuds de calcul AWS PCS pour utiliser le modèle de lancement. Pour plus d'informations, consultez le guide de l'utilisateur de AWS PCS Compute Node Groups.

    - Définissez le groupe `purchaseOption` de nœuds de calcul sur`ONDEMAND`.

Exemple : réserver et utiliser des instances hpc6a.48xlarge avec un ODCR ciblé

Cet exemple de commande crée un ODCR ciblé pour 32 instances hpc6a.48xlarge. Pour lancer les instances réservées dans un groupe de placement, ajoutez `--placement-group-arn` à la commande. Vous pouvez définir une date de fin avec `--end-date` et`--end-date-type`, sinon, la réservation se poursuivra jusqu'à ce qu'elle soit annulée manuellement.

```
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
```

Le résultat de cette commande sera un ARN pour le nouvel ODCR. L'ID ODCR peut être récupéré à partir de l'ARN `"arn:aws:ec2:us-east-2:123456789012:capacity-reservation/ODCR-ID"` ou à l'aide d'Amazon DescribeCapacityReservations EC2.

Association ODCR directe : ajoutez l'identifiant ODCR au modèle de lancement. Voici un exemple de modèle de lancement qui fait référence à l'identifiant ODCR.

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

Association de groupes de réservation de capacité : créez un groupe de réservation de capacité et ajoutez-le au modèle de lancement. La commande suivante crée un groupe de réservation de capacité nomméEXAMPLE-CR-GROUP.

```
aws resource-groups create-group \
```

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

La commande suivante ajoute l'ODCR au groupe de réservation de capacité.

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

Une fois l'ODCR créé et ajouté à un groupe de réservation de capacité, il peut désormais être connecté à un groupe de nœuds de calcul AWS PCS en l'ajoutant à un modèle de lancement. Voici un exemple de modèle de lancement qui fait référence au groupe de réservation de capacité.

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

Enfin, créez ou mettez à jour un groupe de nœuds de calcul AWS PCS pour utiliser les instances hpc6a.48xlarge et utilisez le modèle de lancement qui fait référence à l'ODCR. Pour un groupe de nœuds statique, définissez les instances minimale et maximale en fonction de la taille de la réservation (32). Pour un groupe de nœuds dynamique, définissez le nombre minimum d'instances sur 0 et le maximum sur la taille d'instance souhaitée.

Cet exemple est une implémentation simple d'un ODCR unique configuré pour un groupe de nœuds de calcul. Mais AWS PCS prend en charge de nombreux autres modèles. Par exemple, vous pouvez subdiviser un grand groupe ODCR ou de réservation de capacité entre plusieurs groupes de nœuds de calcul. Vous pouvez également utiliser ODCRs le compte AWS créé et partagé avec le vôtre.

Pour plus d'informations, consultez la section Réservations de capacité à la demande et blocs de capacité pour le ML dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Utilisation des blocs de capacité Amazon EC2 pour le ML avec PCS AWS

Amazon EC2 Capacity Blocks for ML est une option d'achat Amazon EC2 qui vous permet de payer à l'avance pour réserver des instances de calcul accéléré basées sur un GPU à une date et à une

heure spécifiques afin de prendre en charge des charges de travail de courte durée. Les instances qui s'exécutent au sein d'un bloc de capacité sont automatiquement placées à proximité les unes des autres dans Amazon EC2 UltraClusters, pour une mise en réseau non bloquante à faible latence, à l'échelle du pétabit. Pour plus d'informations, consultez Capacity Blocks for ML dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Vous pouvez utiliser un modèle de lancement pour que AWS PCS utilise un bloc de capacité lorsqu'il lance des instances pour un groupe de nœuds de calcul.

> **Note**
>
> AWS PCS a introduit le support pour les blocs de capacité depuis la version 24.05 de Slurm.

Limitations

- AWS PCS prend uniquement en charge les blocs de capacité avec les familles d'instances P5en, P5e, P5 et P4d.
- Vous ne pouvez associer un groupe de nœuds de calcul qu'à un seul bloc de capacité à la fois.
- Vous ne pouvez pas associer un groupe de nœuds de calcul à un groupe de réservation de capacité combinant plusieurs blocs de capacité.
- Les blocs de capacité doivent être à `active` l'état `scheduled` ou pour être utilisés avec le AWS PCS. Vous ne pouvez pas utiliser les blocs de capacité dans d'autres États, tels que`payment-failed`. Pour plus d'informations, consultez la section Afficher les blocs de capacité dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Expiration du bloc de capacité

Les blocs de capacité sont limités à une plage de dates et d'heures spécifiques. Lorsqu'un bloc de capacité expire :

- Le groupe de nœuds de calcul associé à ce bloc de capacité continue d'exister et reste associé aux mêmes files d'attente.
- Toutes les instances du groupe de nœuds de calcul sont mises hors service et les tâches actives risquent d'échouer, selon vos paramètres Slurm.
- AWS PCS ne peut pas lancer de nouvelles instances dans le groupe de nœuds de calcul.

- Toutes les tâches mises en file d'attente ou récemment soumises restent en attente jusqu'à ce qu'un autre groupe de nœuds de calcul soit attaché à la file d'attente ou que vous mettiez à jour le groupe de nœuds de calcul pour utiliser un nouveau modèle de lancement spécifiant un nouveau bloc de capacité.

Configuration d'un groupe de nœuds de calcul AWS PCS pour utiliser un bloc de capacité

Pour associer un bloc de capacité à un groupe de nœuds de calcul

1. Créez un modèle de EC2 lancement Amazon pour AWS PCS qui spécifie votre bloc de capacité. Pour plus d'informations sur la création d'un modèle de lancement pour AWS PCS, consultezUtilisation des modèles de lancement Amazon EC2 avec PCS AWS.

   Votre modèle de lancement doit inclure :

   - La valeur `MarketType` de `InstanceMarketOptions` doit être définie sur`capacity-block`.
   - A `CapacityReservationSpecification` avec un `CapacityReservationId`
   - Une valeur valide `InstanceType` correspondant au type d'instance du Capacity Block que vous avez acheté.
2. Créez un groupe de nœuds de calcul qui utilise le modèle de lancement. Pour de plus amples informations, veuillez consulter Création d'un groupe de nœuds de calcul dans AWS PCS. Vous pouvez également mettre à jour un groupe de nœuds de calcul existant pour utiliser le modèle de lancement. Pour de plus amples informations, veuillez consulter Mise à jour d'un groupe de nœuds de calcul AWS PCS.

   Lorsque vous créez ou mettez à jour le groupe de nœuds de calcul :

   - L'identité IAM que vous utilisez pour créer ou mettre à jour le groupe de nœuds de calcul doit disposer de l'autorisation suivante :

     ```
     ec2:DescribeCapacityReservations
     ```

     Pour de plus amples informations, veuillez consulter Autorisations minimales pour les AWS PCS.
   - Le bloc de capacité doit être à `active` l'état `scheduled` ou.

- Définissez le groupe `purchaseOption` de nœuds de calcul sur`CAPACITY_BLOCK`.
- La taille `maxInstanceCount` du groupe de nœuds de calcul ne doit pas dépasser la taille du bloc de capacité.
- La zone de disponibilité du groupe de nœuds de calcul doit correspondre à l'une des zones de disponibilité de sous-réseau du groupe de nœuds de calcul.

> **Important**
>
> Vous ne pouvez pas modifier le type d'instance d'un groupe de nœuds de calcul lorsque vous le mettez à jour. Vous ne pouvez utiliser un bloc de capacité qu'avec le même type d'instance que le groupe de nœuds de calcul. Si vous souhaitez utiliser un bloc de capacité avec un autre type d'instance, vous devez créer un nouveau groupe de nœuds de calcul.

Questions fréquemment posées sur l'utilisation des blocs de capacité avec les AWS PCS

Je viens de payer un bloc de capacité et j'ai immédiatement essayé de l'utiliser avec un AWS PCS, mais la création d'un groupe de nœuds de calcul a échoué. Que s'est-il passé ?

Votre bloc de capacité n'est peut-être pas dans `active` l'état « `scheduled` or ». Réessayez une fois que le bloc de capacité est `scheduled` ou`active`.

J'utilise un bloc de capacité dans AWS PCS et j'ai acheté une extension avant son expiration. Comment puis-je continuer à l'utiliser dans AWS PCS ?

Vous n'avez rien à faire pour continuer à utiliser le bloc de capacité dans AWS PCS. La date de fin de votre bloc de capacité est mise à jour une fois que le paiement de votre extension a été effectué avec succès. Tant que votre bloc de capacité n'expire pas, le groupe de nœuds de calcul continue de fonctionner. Si le paiement de votre extension échoue, votre bloc de capacité est conservé `active` et le groupe de nœuds de calcul fonctionne jusqu'à ce que le bloc de capacité expire à sa date de fin initiale.

Qu'advient-il de mes tâches en attente et en cours d'exécution si mon bloc de capacité expire ?

Les tâches en file d'attente qui n'ont pas démarré avant l'expiration du bloc de capacité restent en attente jusqu'à ce que vous attachiez un autre groupe de nœuds de calcul à la file d'attente ou que vous mettiez à jour le groupe de nœuds de calcul avec un nouveau bloc de capacité. Vous

pouvez toujours ajouter des tâches à la file d'attente. Vos paramètres Slurm ont une incidence sur les tâches actives. Par défaut, les tâches actives sont automatiquement mises en file d'attente, mais elles peuvent comporter des erreurs ou échouer.

Mon bloc de capacité a expiré. Dois-je faire quelque chose ?

Tu n'as rien à faire. Vous pouvez consulter la console Amazon EC2 pour connaître l'état de vos réservations de capacité EC2. Lorsqu'un bloc de capacité expire, le groupe de nœuds de calcul associé à ce bloc de capacité continue d'exister et de gérer les mêmes files d'attente. Le groupe de nœuds de calcul ne possède aucune instance pour exécuter des tâches. Vous pouvez supprimer le groupe de nœuds de calcul ou le dissocier des files d'attente pour empêcher les utilisateurs de soumettre des tâches qui ne seront pas exécutées.

Je souhaite utiliser un nouveau bloc de capacité avec mon groupe de nœuds de calcul AWS PCS. Que dois-je faire ?

Nous vous recommandons de créer un nouveau groupe de nœuds de calcul pour utiliser le nouveau bloc de capacité. Pour de plus amples informations, veuillez consulter Configuration d'un groupe de nœuds de calcul AWS PCS pour utiliser un bloc de capacité.

Comment puis-je partager un bloc de capacité entre les clusters et les services ?

Vous pouvez répartir un bloc de capacité sur plusieurs clusters et services. Par exemple, pour diviser un bloc de capacité avec 64 `p5.48xlarge` instances avec 20 nœuds sur PCS-Cluster-1, 16 nœuds sur PCS-Cluster-2 et les nœuds restants pour d'autres services, définissez les deux et sur 20 pour PCS-Cluster-1 `minInstanceCount` et 16 `maxInstanceCount` pour PCS-Cluster-2.

Puis-je utiliser plus d'un bloc de capacité ou une capacité combinée avec un groupe de nœuds de calcul ?

Non Un seul bloc de capacité peut être associé à un seul groupe de nœuds de calcul. AWS Le PCS ne prend pas en charge les groupes de réservation de capacité qui combinent plusieurs blocs de capacité.

Comment savoir quand mes blocs de capacité commencent ou expirent ?

Indépendamment de AWS PCS, Amazon EC2 envoie un `Capacity Block Reservation Delivered` événement EventBridge lorsque la réservation d'un bloc de capacité commence et un `Capacity Block Reservation Expiration Warning` événement 40 minutes avant l'expiration de la réservation du bloc de capacité. Pour plus d'informations, consultez la section Monitor Capacity Blocks using EventBridge dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Comment est-ce que Slurm suit l'état de mon bloc de capacité ?

Vous pouvez courir `sinfo` pour comprendre comment AWS PCS utilise le bloc de capacité. Dans l'exemple de sortie suivant, une file d'attente est associée à un groupe de nœuds de calcul qui exécute 4 instances à partir d'un bloc de `active` capacité. Les nœuds sont dans l'état `idle` Slurm (ils peuvent être utilisés et ne sont pas encore affectés à des tâches).

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

Si les nœuds sont plutôt en `maint` état, vous pouvez courir `scontrol show res` pour voir les détails de la réservation Slurm qui contrôle cet état. Dans l'exemple de sortie suivant, le bloc de capacité `scheduled` indique une date de début future.

```
$ scontrol show res

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

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

   Comment=node-fanout Scheduled
```

Comment puis-je savoir si les erreurs que je reçois lors du lancement de Capacity sont dues au fait que mon bloc de capacité est partagé ?

Consultez les réservations de capacité dans la console Amazon EC2 pour savoir combien d'instances du bloc de capacité sont activement provisionnées. Vérifiez les balises de chaque instance pour savoir quel service ou cluster l'utilise. Par exemple, toutes les instances pour AWS AWS PCS possèdent des balises PCS `aws:pcs:cluster-id = pcs_l0mizqyk5o | aws:pcs:compute-node-group-id = pcs_ic7onkmfqk` qui indiquent à quels clusters et groupes de nœuds de calcul l'instance appartient. Vous pouvez ensuite vérifier si le bloc de capacité est à sa capacité maximale.

Vous utilisez `scontrol show nodes` pour vérifier si un nœud Capacity Block d'un cluster AWS PCS déclenche `ReservationCapacityExceeded` :

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

Lorsque plusieurs groupes de nœuds de calcul sont attachés à la même file d'attente, comment puis-je forcer l'exécution d'une tâche sur des instances basées sur Capacity Block ?

Vous pouvez utiliser les fonctionnalités et les contraintes de Slurm pour verrouiller une tâche sur un certain ensemble de nœuds. Nous vous recommandons de ne pas définir de poids Slurm pour chaque groupe de nœuds de calcul, car cela ne fonctionne qu'avec les nœuds qui ne sont pas dans cet état. `maint`

Paramètres utiles du modèle de lancement

Cette section décrit certains paramètres du modèle de lancement qui peuvent être largement utiles avec AWS PCS.

Activez la CloudWatch surveillance détaillée

Vous pouvez activer la collecte de CloudWatch métriques à un intervalle plus court à l'aide d'un paramètre de modèle de lancement.

AWS Management Console

Sur les pages de console permettant de créer ou de modifier des modèles de lancement, cette option se trouve dans la section Détails avancés. Définissez la CloudWatch surveillance détaillée sur Activer.

YAML

```
Monitoring:
    Enabled: True
```

JSON

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

Pour plus d'informations, consultez Activer ou désactiver la surveillance détaillée de vos instances dans le guide de l'utilisateur Amazon Elastic Compute Cloud pour les instances Linux.

Service de métadonnées d'instance, version 2 (IMDS v2)

L'utilisation d'IMDS v2 avec des instances EC2 améliore considérablement la sécurité et contribue à atténuer les risques potentiels associés à l'accès aux métadonnées des instances dans AWS les environnements.

AWS Management Console

Sur les pages de console permettant de créer ou de modifier des modèles de lancement, cette option se trouve dans la section Détails avancés. Définissez les métadonnées accessibles sur Activé, la version des métadonnées sur V2 uniquement (jeton requis) et la limite de sauts de réponse des métadonnées sur 4.

YAML

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

JSON

```
{
```

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

.

AWS files d'attente PCS

Une file d'attente AWS PCS est une abstraction légère de l'implémentation native d'une file d'attente de travail par le planificateur. Dans le cas de Slurm, une file d'attente AWS PCS est équivalente à une partition Slurm.

Les utilisateurs soumettent les tâches à une file d'attente où elles résident jusqu'à ce qu'elles puissent être planifiées pour s'exécuter sur des nœuds fournis par un ou plusieurs groupes de nœuds de calcul. Un cluster AWS PCS peut comporter plusieurs files d'attente de tâches. Par exemple, vous pouvez créer une file d'attente qui utilise les instances Amazon EC2 On-Demand pour les tâches prioritaires et une autre qui utilise les instances Amazon EC2 Spot pour les tâches peu prioritaires.

Rubriques

- Création d'une file d'attente dans AWS PCS
- Mettre à jour une file d'attente AWS PCS
- Supprimer une file d'attente dans AWS PCS

Création d'une file d'attente dans AWS PCS

Cette rubrique fournit un aperçu des options disponibles et décrit les éléments à prendre en compte lors de la création d'une file d'attente dans AWS PCS.

> **Note**
>
> Vous pouvez configurer des paramètres Slurm personnalisés sur les files d'attente afin de mettre en œuvre des politiques de planification et de gestion des ressources spécifiques aux partitions. Pour de plus amples informations, veuillez consulter Configuration des paramètres personnalisés de Slurm dans PCS AWS.

Prérequis

- Un cluster AWS PCS : les files d'attente ne peuvent être créées qu'en association avec un cluster AWS PCS spécifique.
- Un ou plusieurs groupes de nœuds de calcul AWS PCS : une file d'attente doit être associée à au moins un groupe de nœuds de calcul AWS PCS.

Pour créer une file d'attente dans AWS PCS

Vous pouvez créer une file d'attente à l'aide du AWS Management Console ou du AWS CLI.

AWS Management Console

Pour créer une file d'attente à l'aide de la console

1. Ouvrez la console AWS PCS.
2. Sélectionnez le cluster pour la file d'attente. Accédez à Files d'attente, puis choisissez Créer une file d'attente.
3. Dans la section Configuration de la file d'attente, indiquez les valeurs suivantes :
   a. Nom de la file d'attente : nom de votre file d'attente. Un nom ne peut contenir que des caractères alphanumériques (sensibles à la casse) et des traits d'union. Il doit commencer par un caractère alphabétique et ne doit pas comporter plus de 25 caractères. Le nom doit être unique au sein du cluster.
   b. Groupes de nœuds de calcul : sélectionnez un ou plusieurs groupes de nœuds de calcul pour desservir cette file d'attente. Un groupe de nœuds de calcul peut être associé à plusieurs files d'attente.
4. (Facultatif) Dans la section Paramètres supplémentaires du planificateur, vous pouvez ajouter des paires de nom et de valeur des paramètres pour configurer des paramètres supplémentaires de Slurm. Pour obtenir la liste complète des paramètres pris en charge, consultezParamètres Slurm personnalisés pour AWS les files d'attente PCS.
5. (Facultatif) Sous Balises, ajoutez des balises à votre file d'attente AWS PCS
6. Choisissez Créez une file d'attente. Le champ Status affichera Creating tandis que AWS PCS crée la file d'attente. La création d'une file d'attente peut prendre plusieurs minutes.

Étape suivante recommandée

- Soumettez une tâche à votre nouvelle file d'attente.

AWS CLI

Pour créer une file d'attente à l'aide de AWS CLI

Utilisez la commande suivante pour créer votre file d'attente. Procédez aux remplacements suivants :

1. Remplacez *region-code* par la AWS région du cluster. Par exemple, `us-east-1`.
2. Remplacez *my-queue* par le nom de votre file d'attente. Un nom ne peut contenir que des caractères alphanumériques (sensibles à la casse) et des traits d'union. Il doit commencer par un caractère alphabétique et ne doit pas comporter plus de 25 caractères. Le nom doit être unique au sein du cluster.
3. Remplacez *my-cluster* par le nom ou l'ID de votre cluster.
4. *compute-node-group-id*Remplacez-le par l'ID du groupe de nœuds de calcul pour desservir la file d'attente. Par exemple, `pcs_abcdef12345`.

> **Note**
>
> Lorsque vous créez une file d'attente, vous devez fournir l'ID du groupe de nœuds de calcul et non son nom.

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

Example— Création d'une file d'attente avec des paramètres Slurm personnalisés

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

Pour de plus amples informations, veuillez consulter Paramètres Slurm personnalisés pour AWS les files d'attente PCS.

La création de la file d'attente peut prendre plusieurs minutes. Vous pouvez demander l'état de votre file d'attente à l'aide de la commande suivante. Vous ne serez pas en mesure de soumettre des tâches à la file d'attente tant que son statut n'aura pas été atteint`ACTIVE`.

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

Étape suivante recommandée

- Soumettre une tâche à votre nouvelle file d'attente

Mettre à jour une file d'attente AWS PCS

Cette rubrique fournit une vue d'ensemble des options disponibles et décrit les éléments à prendre en compte lors de la mise à jour d'une file d'attente AWS PCS. Pour plus d'informations sur les paramètres personnalisés de Slurm, consultez. Paramètres Slurm personnalisés pour AWS les files d'attente PCS

Considérations relatives à la mise à jour d'une file AWS PCS

Les mises à jour de file d'attente n'auront aucun impact sur les tâches en cours, mais le cluster risque de ne pas être en mesure d'accepter de nouvelles tâches pendant la mise à jour de la file d'attente.

Pour mettre à jour une file d'attente AWS PCS

Vous pouvez utiliser le AWS Management Console ou AWS CLI pour mettre à jour une file d'attente.

AWS Management Console

Pour mettre à jour une file d'attente

1. Ouvrez la console AWS PCS à l'adresse `https://console.aws.amazon.com/pcs/home#/clusters`
2. Sélectionnez le cluster dans lequel vous souhaitez mettre à jour une file d'attente.

3. Accédez aux files d'attente, accédez à la file que vous souhaitez mettre à jour, puis sélectionnez Modifier.
4. Dans la section de configuration de la file d'attente, mettez à jour l'une des valeurs suivantes :
   - Groupes de nœuds : ajoutez ou supprimez des groupes de nœuds de calcul associés à la file d'attente.
   - Paramètres supplémentaires du planificateur : ajoutez, modifiez ou supprimez les paramètres Slurm personnalisés pour la file d'attente. Pour de plus amples informations, veuillez consulter Paramètres Slurm personnalisés pour AWS les files d'attente PCS.
   - Balises : ajoutez ou supprimez des balises pour la file d'attente.
5. Choisissez Mettre à jour. Le champ État affichera la mise à jour pendant que les modifications sont appliquées.

> **Important**
>
> Les mises à jour des files d'attente peuvent prendre plusieurs minutes.

AWS CLI

Pour mettre à jour une file d'attente

1. Mettez à jour votre file d'attente avec la commande suivante. Avant d'exécuter la commande, effectuez les remplacements suivants :
   a. *region-code*Remplacez-le par Région AWS celui dans lequel vous souhaitez créer votre cluster.
   b. Remplacez *my-queue* par le nom ou par celui `computeNodeGroupId` de votre file d'attente.
   c. Remplacez *my-cluster* par le nom ou `clusterId` celui de votre cluster.
   d. Pour modifier les associations de groupes de nœuds de calcul, fournissez une liste mise à jour pour`--compute-node-group-configurations`.
      - Par exemple, pour ajouter un deuxième groupe de nœuds de calcul `computeNodeGroupExampleID2` :

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

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

Example— Mise à jour d'une file d'attente avec des paramètres Slurm personnalisés

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

Pour de plus amples informations, veuillez consulter Paramètres Slurm personnalisés pour AWS les files d'attente PCS.

2. La mise à jour de la file d'attente peut prendre plusieurs minutes. Vous pouvez demander l'état de votre file d'attente à l'aide de la commande suivante. Vous ne pourrez pas soumettre de tâches à la file d'attente tant que son statut n'aura pas été atteint`ACTIVE`.

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

Prochaines étapes recommandées

- Soumettez une tâche à votre file d'attente mise à jour.

Supprimer une file d'attente dans AWS PCS

Cette rubrique explique comment supprimer une file d'attente dans AWS PCS.

Considérations relatives à la suppression d'une file d'attente

- Si des tâches sont en cours d'exécution dans la file d'attente, elles seront interrompues par le planificateur lorsque la file d'attente sera supprimée. Les tâches en attente dans la file d'attente seront annulées. Envisagez d'attendre que les tâches de la file d'attente soient terminées ou de stop/cancel les exécuter manuellement à l'aide des commandes natives du planificateur (comme `scancel` pour Slurm).

Supprimer la file d'attente

Vous pouvez utiliser le AWS Management Console ou AWS CLI pour supprimer une file d'attente.

AWS Management Console

Pour supprimer une file d'attente

1. Ouvrez la console AWS PCS.
2. Sélectionnez le cluster de la file d'attente.
3. Accédez à Files d'attente et sélectionnez la file à supprimer.
4. Sélectionnez Delete (Supprimer).
5. Le champ État s'affiche`Deleting`. Cela peut prendre plusieurs minutes.

> **Note**
>
> Vous pouvez utiliser les commandes natives de votre planificateur pour confirmer que la file d'attente est supprimée. Par exemple, utilisez `sinfo` ou `squeue` pour Slurm.

AWS CLI

Pour supprimer une file d'attente

- Utilisez la commande suivante pour supprimer une file d'attente, avec les remplacements suivants :
  - *`region-code`*Remplacez-le par celui dans lequel se trouve Région AWS votre cluster.
  - Remplacez *`my-queue`* par le nom ou l'ID de votre file d'attente.

- Remplacez *my-cluster* par le nom ou l'ID de votre cluster.

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

La suppression de la file d'attente peut prendre plusieurs minutes.

> **Note**
>
> Vous pouvez utiliser les commandes natives de votre planificateur pour confirmer que la file d'attente est supprimée. Par exemple, utilisez `sinfo` ou `squeue` pour Slurm.

AWS Nœuds de connexion PCS

Un cluster AWS PCS a généralement besoin d'au moins un nœud de connexion pour prendre en charge l'accès interactif et la gestion des tâches. Un moyen d'y parvenir consiste à utiliser un groupe de nœuds de calcul AWS PCS statique configuré pour la capacité du nœud de connexion. Vous pouvez également configurer une instance EC2 autonome pour qu'elle fasse office de nœud de connexion.

Rubriques

- Utilisation d'un groupe de nœuds de calcul AWS PCS pour fournir des nœuds de connexion
- Utilisation d'instances autonomes comme nœuds de connexion AWS PCS
- Connexion d'un nœud de connexion autonome à plusieurs clusters dans AWS PCS

Utilisation d'un groupe de nœuds de calcul AWS PCS pour fournir des nœuds de connexion

Cette rubrique fournit une vue d'ensemble des options de configuration suggérées et décrit les éléments à prendre en compte lorsque vous utilisez un groupe de nœuds de calcul AWS PCS pour fournir un accès permanent et interactif à votre cluster.

Création d'un groupe de nœuds de calcul AWS PCS pour les nœuds de connexion

Sur le plan opérationnel, cela n'est pas très différent de la création d'un groupe de nœuds de calcul normal. Cependant, certains choix de configuration clés sont à effectuer :

- Définissez une configuration de dimensionnement statique pour au moins une instance EC2 dans le groupe de nœuds de calcul.
- Choisissez l'option d'achat à la demande pour éviter que vos instances ne soient récupérées.
- Choisissez un nom informatif pour le groupe de nœuds de calcul, tel que login.
- Si vous souhaitez que les instances du nœud de connexion soient accessibles en dehors de votre VPC, pensez à utiliser un sous-réseau public.
- Si vous avez l'intention d'autoriser l'accès SSH, le modèle de lancement doit disposer d'un groupe de sécurité qui expose le port SSH aux adresses IP de votre choix.

- Le profil d'instance IAM ne doit comporter que les autorisations AWS que vous souhaitez attribuer à vos utilisateurs finaux. Consultez Profils d'instance IAM pour AWS Parallel Computing Service pour plus de détails.
- Envisagez d'autoriser AWS Systems Manager Session Manager à gérer vos instances de connexion.
- Envisagez de restreindre l'accès aux informations d'identification AWS de l'instance aux seuls utilisateurs administratifs
- Sélectionnez des types d'instance moins coûteux que pour les groupes de nœuds de calcul classiques, car le ou les nœuds de connexion fonctionneront en continu.
- Utilisez la même AMI (ou une AMI dérivée) que pour vos autres groupes de nœuds de calcul afin de garantir que le même logiciel est installé sur toutes les instances. Pour plus d'informations sur la personnalisation AMIs, voir Amazon Machine Images (AMIs) pour AWS PC
- Configurez le même système de fichiers réseau (Amazon EFS, Amazon FSx for Lustre, etc.) que les montages sur vos nœuds de connexion et sur vos instances de calcul. Pour de plus amples informations, veuillez consulter Utilisation de systèmes de fichiers réseau avec AWS PCS.

Accédez à vos nœuds de connexion

Une fois que votre nouveau groupe de nœuds de calcul atteint le statut ACTIF, vous pouvez trouver les instances EC2 qu'il a créées et vous y connecter. Pour de plus amples informations, veuillez consulter Recherche d'instances de groupes de nœuds de calcul dans AWS PCS.

Mise à jour d'un groupe de nœuds de calcul AWS PCS pour les nœuds de connexion

Vous pouvez mettre à jour un groupe de nœuds de connexion à l'aide de UpdateComputeNodeGroup. Dans le cadre du processus de mise à jour du groupe de nœuds, les instances en cours d'exécution seront remplacées. Notez que cela interrompra toutes les sessions utilisateur ou tous les processus actifs sur l'instance. Les jobs Slurm en cours ou en file d'attente ne seront pas affectés. Pour de plus amples informations, veuillez consulter Mise à jour d'un groupe de nœuds de calcul AWS PCS.

Vous pouvez également modifier le modèle de lancement utilisé par votre groupe de nœuds de calcul. Vous devez l'utiliser UpdateComputeNodeGroup pour appliquer le modèle de lancement mis à jour au groupe de nœuds de calcul. Les nouvelles instances EC2 lancées dans le groupe de nœuds

de calcul utilisent le modèle de lancement mis à jour. Pour de plus amples informations, veuillez consulter Utilisation des modèles de lancement Amazon EC2 avec PCS AWS.

Suppression d'un groupe de nœuds de calcul AWS PCS pour les nœuds de connexion

Vous pouvez mettre à jour un groupe de nœuds de connexion à l'aide du mécanisme de suppression du groupe de nœuds de calcul dans AWS PCS. Les instances en cours d'exécution seront interrompues dans le cadre de la suppression du groupe de nœuds. Notez que cela interrompra toutes les sessions utilisateur ou tous les processus actifs sur l'instance. Les jobs Slurm en cours ou en file d'attente ne seront pas affectés. Pour de plus amples informations, veuillez consulter Suppression d'un groupe de nœuds de calcul dans AWS PCS.

Utilisation d'instances autonomes comme nœuds de connexion AWS PCS

Vous pouvez configurer des instances EC2 indépendantes pour interagir avec le planificateur AWS Slurm d'un cluster PCS. Cela est utile pour créer des nœuds de connexion, des postes de travail ou des hôtes de gestion de flux de travail dédiés qui fonctionnent avec des clusters AWS PCS mais fonctionnent en dehors de la gestion AWS PCS. Pour ce faire, chaque instance autonome doit :

1. Installez une version compatible du logiciel Slurm.
2. Être capable de se connecter au point de terminaison Slurmctld du cluster AWS PCS.
3. Configurez correctement les démons Slurm Auth et Cred Kiosk (`sackd`) avec le point de terminaison et le secret du cluster AWS PCS. Pour plus d'informations, consultez sackd dans la documentation de Slurm.

Ce didacticiel vous aide à configurer une instance indépendante qui se connecte à un cluster AWS PCS.

Table des matières

- Étape 1 — Récupérez l'adresse et le secret du cluster AWS PCS cible
- Étape 2 — Lancer une instance EC2
- Étape 3 — Installation de Slurm sur l'instance
- Étape 4 — Récupérez et stockez le secret du cluster

- Étape 5 — Configuration de la connexion au cluster AWS PCS
- Étape 6 — (Facultatif) Testez la connexion

Étape 1 — Récupérez l'adresse et le secret du cluster AWS PCS cible

Récupérez les détails du cluster AWS PCS cible à l' AWS CLI aide de la commande suivante. Avant d'exécuter la commande, effectuez les remplacements suivants :

- *region-code*Remplacez-le par celui Région AWS où s'exécute le cluster cible.
- Remplacer *cluster-ident* par le nom ou l'identifiant du cluster cible

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

La commande renverra une sortie similaire à celle de cet exemple.

```
{
    "cluster": {
        "name": "get-started",
        "id": "pcs_123456abcd",
        "arn": "arn:aws:pcs:us-east-1:111122223333:cluster/pcs_123456abcd",
        "status": "ACTIVE",
        "createdAt": "2024-12-17T21:03:52+00:00",
        "modifiedAt": "2024-12-17T21:03:52+00:00",
        "scheduler": {
            "type": "SLURM",
            "version": "25.05"
        },
        "size": "SMALL",
        "slurmConfiguration": {
            "authKey": {
                "secretArn": "arn:aws:secretsmanager:us-east-1:111122223333:secret:pcs!
slurm-secret-pcs_123456abcd-a12ABC",
                "secretVersion": "ef232370-d3e7-434c-9a87-ec35c1987f75"
            }
        },
        "networking": {
            "subnetIds": [
                "subnet-0123456789abcdef0"
            ],
            "securityGroupIds": [
```

```
                "sg-0123456789abcdef0"
            ]
        },
        "endpoints": [
            {
                "type": "SLURMCTLD",
                "privateIpAddress": "10.3.149.220",
                "port": "6817"
            }
        ]
    }
}
```

Dans cet exemple, le point de terminaison du contrôleur Slurm du cluster possède une adresse IP de `10.3.149.220` et s'exécute sur le port. `6817` Il `secretArn` sera utilisé dans les étapes ultérieures pour récupérer le secret du cluster. L'adresse IP et le port seront utilisés ultérieurement pour configurer le `sackd` service.

Étape 2 — Lancer une instance EC2

Pour lancer une instance EC2

1. Ouvrez la console Amazon EC2.
2. Dans le volet de navigation, choisissez Instances, puis Launch Instances (Lancer des instances) pour ouvrir le nouvel assistant de lancement d'instance.
3. (Facultatif) Dans la section Nom et balises, saisissez un nom pour l'instance, par exemple`PCS-LoginNode`. Le nom est attribué à l'instance en tant qu'identification de ressource (`Name=PCS-LoginNode`).
4. Dans la section Images de l'application et du système d'exploitation, sélectionnez une AMI pour l'un des systèmes d'exploitation pris en charge par AWS PCS. Pour de plus amples informations, veuillez consulter Systèmes d'exploitation pris en charge.
5. Dans la section Type d'instance, sélectionnez un type d'instance pris en charge. Pour de plus amples informations, veuillez consulter Types d'instance pris en charge.
6. Dans la section Paire de clés, sélectionnez la paire de clés SSH à utiliser pour l'instance.
7. Dans la section Paramètres réseau :

    - Choisissez Modifier.

        i. Sélectionnez le VPC de votre cluster AWS PCS.

ii. Pour Firewall (security groups) (Pare-feu (groupes de sécurité), choisissez Select existing security group (Sélectionner un groupe de sécurité existant).

A. Sélectionnez un groupe de sécurité qui autorise le trafic entre l'instance et le contrôleur Slurm du cluster AWS PCS cible. Pour de plus amples informations, veuillez consulter Exigences et considérations relatives aux groupes de sécurité.

B. (Facultatif) Sélectionnez un groupe de sécurité qui autorise l'accès SSH entrant à votre instance.

8. Dans la section Stockage, configurez les volumes de stockage selon vos besoins. Assurez-vous de configurer suffisamment d'espace pour installer les applications et les bibliothèques afin d'activer votre cas d'utilisation.
9. Sous Avancé, choisissez un rôle IAM qui autorise l'accès au secret du cluster. Pour de plus amples informations, veuillez consulter Obtenez le secret du cluster Slurm.
10. Dans le volet Résumé, choisissez Launch instance.

Étape 3 — Installation de Slurm sur l'instance

Lorsque l'instance est lancée et devient active, connectez-vous à celle-ci en utilisant le mécanisme de votre choix. Utilisez le programme d'installation de Slurm fourni par AWS pour installer Slurm sur l'instance. Pour de plus amples informations, veuillez consulter Installateur Slurm.

Téléchargez le programme d'installation de Slurm, décompressez-le et utilisez le `installer.sh` script pour installer Slurm. Pour de plus amples informations, veuillez consulter Étape 3 — Installation de Slurm.

Étape 4 — Récupérez et stockez le secret du cluster

Ces instructions nécessitent le AWS CLI. Pour plus d'informations, voir Installation ou mise à jour vers la dernière version du AWS CLI dans le guide de AWS Command Line Interface l'utilisateur de la version 2.

Enregistrez le secret du cluster à l'aide des commandes suivantes.

- Créez le répertoire de configuration pour Slurm.

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

- Récupérez, décodez et stockez le secret du cluster. Avant d'exécuter cette commande, *region-code* remplacez-la par la région dans laquelle s'exécute le cluster cible et *secret-arn* remplacez-la par la valeur `secretArn` récupérée à l'étape 1.

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

> **Warning**
>
> Dans un environnement multi-utilisateurs, tout utilisateur ayant accès à l'instance peut être en mesure de récupérer le secret du cluster s'il peut accéder au service de métadonnées d'instance (IMDS). Cela pourrait à son tour leur permettre de se faire passer pour d'autres utilisateurs. Envisagez de restreindre l'accès à l'IMDS aux utilisateurs root ou administratifs uniquement. Vous pouvez également envisager d'utiliser un mécanisme différent qui ne repose pas sur le profil de l'instance pour récupérer et configurer le secret.

- Définissez la propriété et les autorisations sur le fichier clé de Slurm.

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

> **Note**
>
> La clé Slurm doit appartenir à l'utilisateur et au groupe sous lesquels le `sackd` service s'exécute.

Étape 5 — Configuration de la connexion au cluster AWS PCS

Pour établir une connexion au cluster AWS PCS, lancez-le `sackd` en tant que service système en suivant ces étapes.

> **Note**
>
> Si vous utilisez Slurm 25.05 ou une version ultérieure, vous pouvez utiliser un script pour configurer votre nœud de connexion afin qu'il se connecte à plusieurs clusters. Pour de plus amples informations, veuillez consulter Connexion d'un nœud de connexion autonome à plusieurs clusters dans AWS PCS.

1. Configurez le fichier d'environnement du `sackd` service à l'aide de la commande suivante. Avant d'exécuter la commande, remplacez *`ip-address`* et par *`port`* les valeurs extraites des points de terminaison à l'étape 1.

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

2. Créez un fichier `systemd` de service pour gérer le `sackd` processus.

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

[Service]
Type=notify
EnvironmentFile=/etc/sysconfig/sackd
User=slurm
Group=slurm
RuntimeDirectory=slurm
RuntimeDirectoryMode=0755
ExecStart=/opt/aws/pcs/scheduler/slurm-25.05/sbin/sackd --systemd \$SACKD_OPTIONS
ExecReload=/bin/kill -HUP \$MAINPID
KillMode=process
LimitNOFILE=131072
LimitMEMLOCK=infinity
LimitSTACK=infinity

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

3. Définissez la propriété du fichier `sackd` de service.

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

4. Activez le `sackd` service.

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

5. Lancez le service `sackd`.

```
sudo systemctl start sackd
```

Étape 6 — (Facultatif) Testez la connexion

Vérifiez que le `sackd` service est en cours d'exécution. Vous trouverez ci-après un exemple de sortie. S'il y a des erreurs, elles apparaissent généralement ici.

```
[root@ip-10-3-27-112 ~]# systemctl status sackd
[x] sackd.service - Slurm auth and cred kiosk daemon
   Loaded: loaded (/etc/systemd/system/sackd.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2024-12-17 16:34:55 UTC; 8s ago
 Main PID: 9985 (sackd)
   CGroup: /system.slice/sackd.service
           ##9985 /opt/aws/pcs/scheduler/slurm-25.05/sbin/sackd --systemd --conf-
server=10.3.149.220:6817

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

Vérifiez que les connexions au cluster fonctionnent à l'aide des commandes du client Slurm telles que `sinfo` et. `squeue` Voici un exemple de sortie de`sinfo`.

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

Vous devriez également être en mesure de soumettre des offres d'emploi. Par exemple, une commande similaire à cet exemple lancerait une tâche interactive sur un nœud du cluster.

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

Connexion d'un nœud de connexion autonome à plusieurs clusters dans AWS PCS

Le `pcs-multi-cluster-login-configure.sh` script fournit un moyen automatique de configurer plusieurs `sackd` démons Slurm sur un seul nœud de connexion autonome. Il permet au nœud de connexion de communiquer avec plusieurs clusters. Le script automatise les opérations suivantes :

- Utilise les actions de l'API AWS PCS pour obtenir des informations sur le cluster
- Demande de saisie de la clé d'authentification Slurm codée en base64
- Crée un fichier Slurm JWKS avec une clé d'authentification du cluster
- Configure le `sackd` service avec les points de terminaison et les ports du cluster
- Crée un fichier `systemd` de service pour un démon spécifique au cluster `sackd`
- Génère un script d'activation pour la configuration de l'environnement du cluster
- Active et démarre le `sackd` service

> **Note**
>
> Ce script nécessite la version 25.05 ou ultérieure de Slurm.

Slurm doit déjà être installé sur l'instance (équivalent à l'étape 3 du processus manuel). L'instance doit être en mesure d'atteindre les points de terminaison du cluster cible. Le script exécute les opérations équivalentes des étapes 4 et 5 du processus de configuration manuelle. Il obtient automatiquement les informations du cluster, configure le `sackd` service, crée les fichiers de `systemd` service nécessaires et crée un script d'activation que les utilisateurs peuvent utiliser pour configurer leur environnement shell pour l'interaction avec le cluster.

Rubriques

- Conditions préalables pour le script de configuration du nœud de connexion multi-clusters AWS PCS
- AWS Code de script de configuration du nœud de connexion multi-cluster PCS
- Utilisation du script de configuration du nœud de connexion multi-clusters AWS PCS

Conditions préalables pour le script de configuration du nœud de connexion multi-clusters AWS PCS

Configuration système requise

- Système d'exploitation Linux avec `systemd` support
- Privilèges root pour la configuration du système

Commandes et packages requis

- `bash`— Interprète Shell (version 4.0+)
- `curl`— Pour la AWS récupération des métadonnées IMDS v2
- `jq`— Processeur JSON pour analyser les réponses de AWS l'API
- `aws`— AWS CLI v2 pour exécuter les actions de l'API AWS PCS et pour accéder à Secrets Manager
- `systemctl`— gestion des `systemd` services
- `find`— Utilitaire de recherche de système de fichiers
- `grep`— Correspondance du modèle de texte
- `sed`— Éditeur de flux pour la manipulation de texte
- `sort`— Utilitaire de tri de texte
- `tail`— Affiche les dernières lignes d'un fichier
- `mkdir`— Création d'un répertoire
- `chmod`— Modifie les autorisations des fichiers
- `chown`— Modifie la propriété du fichier
- `ldconfig`— Configuration dynamique de l'éditeur de liens

AWS exigences

- Un cluster AWS PCS qui exécute la version 25.05 ou ultérieure de Slurm
- AWS informations d'identification configurées (via un rôle IAM, un fichier d'informations d'identification ou des variables d'environnement)
- Autorisations pour :
  - `pcs:GetCluster`
  - `secretsmanager:GetSecretValue`(si vous utilisez un autre secret)

Utilisateurs et groupes du système

- L'`slurm`utilisateur et le groupe doivent exister sur le système

Installation de Slurm

- Slurm doit être installé au même endroit que les packages d'installation de AWS PCS Slurm :

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

AWS Code de script de configuration du nœud de connexion multi-cluster PCS

Enregistrez le code source suivant dans un fichier portant le nom suivant :

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

Code source du script

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

# AWS PCS Multi-Cluster Standalone Login Node Configuration Script
#
# This script configures AWS Parallel Computing Service (PCS) multi-cluster stand alone
 login nodes
# by setting up the Slurm authentication and credential kiosk daemon (sackd)
```

```
# for connecting to remote PCS clusters.
#
# Prerequisites:
# - AWS CLI configured with appropriate permissions
# - Slurm version 25.05 or later
# - Root privileges for system configuration
# - Network connectivity to AWS PCS endpoints


set -eo pipefail

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

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

# Function to retrieve authentication key
get_auth_key() {
    if [ "$ALTERNATE_SECRET_RETRIEVAL" = "true" ]; then
        echo "Retrieving authentication key from AWS Secrets Manager..." >&2
```

```
        local auth_key_arn=$(echo "$CLUSTER_INFO" | jq -r
 '.cluster.slurmConfiguration.authKey.secretArn')
        local auth_key_version=$(echo "$CLUSTER_INFO" | jq -r
 '.cluster.slurmConfiguration.authKey.secretVersion')

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

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

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

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

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

```
            ((port++))
        fi
    done

    echo "$port"
}

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

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

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

    SACKD_RUNTIME_DIRECTORY="/run/slurm-${CLUSTER_NAME}"
    mkdir -p "${SACKD_RUNTIME_DIRECTORY}"
    chown slurm:slurm "${SACKD_RUNTIME_DIRECTORY}"

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

    SACKD_SERVICE_PATH="/etc/systemd/system/${SACKD_SERVICE_NAME}.service"

    cat << EOF > "$SACKD_SERVICE_PATH"
[Unit]
Description=Slurm auth and cred kiosk daemon
After=network-online.target remote-fs.target
Wants=network-online.target
ConditionPathExists=${SACKD_SERVICE_ENV}
```

```
[Service]
Type=notify
EnvironmentFile=${SACKD_SERVICE_ENV}
User=slurm
Group=slurm
RuntimeDirectory=slurm-${CLUSTER_NAME}
RuntimeDirectoryMode=0755
ExecStart=${SLURM_INSTALL_PATH}/sbin/sackd --systemd \$SACKD_OPTIONS
ExecReload=/bin/kill -HUP \$MAINPID
KillMode=process
LimitNOFILE=131072
LimitMEMLOCK=infinity
LimitSTACK=infinity

[Install]
WantedBy=multi-user.target
EOF

    chown root:root "$SACKD_SERVICE_PATH"
    chmod 0644 "$SACKD_SERVICE_PATH"
    systemctl daemon-reload && systemctl enable "$SACKD_SERVICE_NAME"
    systemctl restart "$SACKD_SERVICE_NAME"

    ACTIVATE_SCRIPT="activate-pcs-${CLUSTER_NAME}"
    cat > "$ACTIVATE_SCRIPT" << EOF
# Activate script for Slurm cluster ${CLUSTER_NAME}

# Add Slurm paths
export PATH="${SLURM_INSTALL_PATH}/bin:\$PATH"
export MANPATH="${SLURM_INSTALL_PATH}/share/man:\$MANPATH"
export LD_LIBRARY_PATH="${SLURM_INSTALL_PATH}/lib:\$LD_LIBRARY_PATH"
ldconfig

# Set Slurm configuration
export SLURM_CONF="/run/slurm-${CLUSTER_NAME}/conf/slurm.conf"
export PCS_CLUSTER_NAME="${CLUSTER_NAME}"
export PCS_CLUSTER_IDENTIFIER="${CLUSTER_IDENTIFIER}"
export PCS_CLUSTER_ID="${CLUSTER_ID}"

echo "Activated PCS cluster environment: ${CLUSTER_NAME}"

# Deactivate function
function deactivate-pcs-${CLUSTER_NAME}() {
```

```
    export PATH="\$(echo "\$PATH" | sed -e "s|${SLURM_INSTALL_PATH}/bin:||g" -e "s|:
${SLURM_INSTALL_PATH}/bin||g" -e "s|^${SLURM_INSTALL_PATH}/bin\$||")"
    export MANPATH="\$(echo "\$MANPATH" | sed -e "s|${SLURM_INSTALL_PATH}/share/man:||
g" -e "s|:${SLURM_INSTALL_PATH}/share/man||g" -e "s|^${SLURM_INSTALL_PATH}/share/man\
$||")"
    export LD_LIBRARY_PATH="\$(echo "\$LD_LIBRARY_PATH" | sed -e "s|
${SLURM_INSTALL_PATH}/lib:||g" -e "s|:${SLURM_INSTALL_PATH}/lib||g" -e "s|^
${SLURM_INSTALL_PATH}/lib\$||")"
    unset SLURM_CONF
    unset PCS_CLUSTER_NAME
    unset PCS_CLUSTER_IDENTIFIER
    unset PCS_CLUSTER_ID
    unset -f deactivate-pcs-${CLUSTER_NAME}
    ldconfig
    echo "Deactivated PCS cluster environment: ${CLUSTER_NAME}"
}

export -f deactivate-pcs-${CLUSTER_NAME}

EOF
}

# Main function
main() {
    # Parse arguments
    CLUSTER_IDENTIFIER=""
    PCS_ENDPOINT_URL=""

    while [ "$1" != "" ]; do
        case $1 in
            --cluster-identifier)
                shift
                CLUSTER_IDENTIFIER="$1"
                ;;
            --endpoint-url)
                shift
                PCS_ENDPOINT_URL="--endpoint-url $1"
                ;;
            -h|--help)
                help
                exit 0
                ;;
            *)
                echo "Invalid argument: $1" >&2
```

```
                usage >&2
                exit 1
                ;;
        esac
        shift
    done

    # Validate required arguments
    if [ -z "$CLUSTER_IDENTIFIER" ]; then
        echo "Error: --cluster-identifier is required" >&2
        usage >&2
        exit 1
    fi

    # Validate running as root
    if [ "$EUID" -ne 0 ]; then
        echo "Error: This script must be run as root" >&2
        exit 1
    fi

    # Validate required commands are available
    for cmd in aws jq curl; do
        if ! command -v "$cmd" &> /dev/null; then
            echo "Error: Required command '$cmd' not found" >&2
            exit 1
        fi
    done

    # Get the region name from IMDS v2 with error handling (try IPv6 first, fallback to
 IPv4)
    echo "Retrieving AWS region from instance metadata..."
    # Try IPv6 IMDS endpoint first (fd00:ec2::254) with fast timeout (1s connect, 2s
 total)
    # If IPv6 fails, fallback to IPv4 IMDS endpoint (169.254.169.254)
    IMDS_ENDPOINT="http://[fd00:ec2::254]"
    if ! TOKEN=$(curl -s -X PUT "${IMDS_ENDPOINT}/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600" --connect-timeout 1 --max-time 2 2>/dev/null); then
        IMDS_ENDPOINT="http://169.254.169.254"
        if ! TOKEN=$(curl -s -X PUT "${IMDS_ENDPOINT}/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600" --max-time 5); then
            echo "Error: Failed to retrieve IMDS token. Ensure this script is running
 on an EC2 instance." >&2
            exit 1
        fi
```

```
    fi

    if ! REGION=$(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" "${IMDS_ENDPOINT}/
latest/dynamic/instance-identity/document" --max-time 5 | jq -r '.region'); then
        echo "Error: Failed to retrieve AWS region from instance metadata" >&2
        exit 1
    fi

    echo "Detected AWS region: $REGION"

    # Retrieve cluster information from AWS PCS
    echo "Retrieving cluster information for: $CLUSTER_IDENTIFIER"
    # shellcheck disable=SC2086
    if ! CLUSTER_INFO=$(aws pcs get-cluster --region "$REGION" --cluster-identifier
 "$CLUSTER_IDENTIFIER" $PCS_ENDPOINT_URL 2>/dev/null); then
        echo "Error: Failed to retrieve cluster information. Check cluster identifier
 and AWS permissions." >&2
        exit 1
    fi

    CLUSTER_ID=$(echo "$CLUSTER_INFO" | jq -r '.cluster.id')
    CLUSTER_NAME="$(echo "$CLUSTER_INFO" | jq -r '.cluster.name')"
    SLURM_VERSION=$(echo "$CLUSTER_INFO" | jq -r '.cluster.scheduler.version')
    SLURM_VERSION=${SLURM_VERSION#Slurm_}

    # Check if Slurm version is >= 25.05
    # shellcheck disable=SC2072
    if [[ "$SLURM_VERSION" < "25.05" ]]; then
        echo "Error: This script requires Slurm version 25.05 or later. Found version:
 $SLURM_VERSION" >&2
        exit 1
    fi

    ENDPOINTS=$(echo "$CLUSTER_INFO" | jq -r '.cluster.endpoints[] | select(.type
 == "SLURMCTLD") | (if .privateIpAddress != "" then .privateIpAddress else "["
 + .ipv6Address + "]" end) + ":" + .port' | tr '\n' ',' | sed 's/,$//')

    # Get BASE64_SLURM_KEY
    BASE64_SLURM_KEY=$(get_auth_key)

    if [ -z "$BASE64_SLURM_KEY" ]; then
        echo "Error: base64 Slurm key cannot be empty" >&2
        exit 1
    fi
```

```
    configure_cluster

    # Final configuration summary
    echo "========================================"
    echo "Configuration completed successfully!"
    echo "========================================"
    echo "Cluster Name: $CLUSTER_NAME"
    echo "Cluster ID: $CLUSTER_ID"
    echo "Slurm Version: $SLURM_VERSION"
    echo "Service Name: $SACKD_SERVICE_NAME"
    echo "SACKD Port: $SACKD_PORT"
    echo
    echo "To activate this cluster environment, run:"
    echo "  source ./$ACTIVATE_SCRIPT"
    echo
    echo "To deactivate this cluster environment, run:"
    echo "  deactivate-pcs-${CLUSTER_NAME}"
    echo
    echo "To check service status:"
    echo "  systemctl status $SACKD_SERVICE_NAME"
    echo
    echo "To view service logs:"
    echo "  journalctl -u $SACKD_SERVICE_NAME -f"
}

# Exit if being sourced for testing
[[ "${BASH_SOURCE[0]}" != "${0}" ]] && return

# Execute main function
main "$@"
```

Utilisation du script de configuration du nœud de connexion multi-clusters AWS PCS

Exécution du script

Pour exécuter le script de configuration

1. Enregistrez le contenu du script dans un fichier nommé :

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

2. Rendez-le exécutable :

```
chmod +x pcs-multi-cluster-login-configure.sh
```

3. Exécutez le script :

```
./pcs-multi-cluster-login-configure.sh --cluster-identifier cluster-name
```

Environnements d'interaction en cluster

Une fois la configuration réussie, le script génère un script d'activation spécifique au cluster dans le répertoire en cours. Le script porte le nom`activate-pcs-cluster-name`. Le script d'activation configure les variables d'environnement et les chemins nécessaires pour interagir avec le cluster cible.

Pour activer un environnement de cluster

- Utilisez la `source` commande pour exécuter le script d'activation

```
source ./activate-pcs-cluster-name
```

Example

```
# Activate cluster environment for cluster 'my-cluster'
source ./activate-pcs-my-cluster

# Now you can use Slurm commands
sinfo
squeue
sbatch my-job.sh
```

À quoi sert le script d'activation

- Définit la variable d'`SLURM_CONF`environnement pour qu'elle pointe vers la configuration du cluster.
- Met `PATH` à jour le pour inclure les fichiers binaires Slurm du cluster.
- Configure les autres variables d'environnement Slurm nécessaires (`MANPATH`,). `LD_LIBRARY_PATH`

- Définit les variables d'identification du cluster AWS PCS.
- Permet une interaction fluide avec le cluster AWS PCS cible.

Pour désactiver un environnement de cluster

- Exécutez la commande de désactivation.

  ```
  deactivate-pcs-cluster-name
  ```

  Example

  ```
  # After activating a cluster
  source ./activate-pcs-my-cluster

  # Work with the cluster
  sinfo

  # Deactivate when done
  deactivate-pcs-my-cluster
  ```

À quoi sert la commande de désactivation

- Restaure la variable d'PATHenvironnement d'origine.
- Désactive les variables d'environnement Slurm spécifiques au cluster.
- Rétablit l'état de pré-activation de l'environnement shell.

> **Note**
>
> L'activation est spécifique à la session et doit provenir de la session shell dans laquelle vous souhaitez interagir avec le cluster.

AWS Mise en réseau PCS

Votre cluster AWS PCS est créé dans un Amazon VPC. Ce chapitre inclut les rubriques suivantes concernant la mise en réseau pour le planificateur et les nœuds de votre cluster.

À l'exception du choix d'un sous-réseau dans lequel lancer les instances, vous devez utiliser des modèles de EC2 lancement pour configurer la mise en réseau des groupes de nœuds de calcul AWS PCS. Pour en savoir plus sur l'utilisation des modèles de lancement, consultez Utilisation des modèles de lancement Amazon EC2 avec PCS AWS.

Rubriques

- AWS Exigences et considérations relatives au PCS, au VPC et aux sous-réseaux
- Création d'un VPC pour votre AWS cluster PCS
- Groupes de sécurité dans AWS PCS
- Plusieurs interfaces réseau dans les AWS PCS
- Groupes de placement pour les instances EC2 dans PCS AWS
- Utilisation d'Elastic Fabric Adapter (EFA) avec PCS AWS

AWS Exigences et considérations relatives au PCS, au VPC et aux sous-réseaux

Lorsque vous créez un cluster AWS PCS, vous spécifiez un VPC, un sous-réseau dans ce VPC. Cette rubrique fournit une vue d'ensemble des exigences et considérations spécifiques au AWS PCS pour le VPC et les sous-réseaux que vous utilisez avec votre cluster. Si vous n'avez pas de VPC à utiliser avec AWS PCS, vous pouvez en créer un à l'aide d'un modèle fourni AWS. CloudFormation Pour plus d'informations VPCs, consultez la section Virtual Private Clouds (VPC) dans le guide de l'utilisateur Amazon VPC.

Exigences et considérations requises pour le VPC

Lorsque vous créez un cluster, le VPC que vous spécifiez doit répondre aux exigences et aux considérations suivantes :

- Le VPC doit disposer d'un nombre suffisant d'adresses IP disponibles pour le cluster, les nœuds et les autres ressources de cluster que vous souhaitez créer. Pour plus d'informations, consultez la

section Adressage IP pour vos sous-réseaux VPCs et sous-réseaux dans le guide de l'utilisateur Amazon VPC.

- Si votre cluster utilise IPv6 :
  - Associez un bloc IPv6 CIDR à votre VPC. Pour plus d'informations, consultez Créer un VPC dans le Guide de l'utilisateur Amazon VPC.

    > **Important**
    >
    > Bien que vous puissiez configurer votre VPC avec les deux IPv4 IPv6, vous ne pouvez choisir qu'un seul type de réseau pour votre cluster.

  - Activez l'attribution automatique d' IPv6 adresses pour vos sous-réseaux.
  - Pour en savoir plus, consultez :
    - IPv6 sur AWS
    - Comprendre l' IPv6 adressage sur AWS et concevoir un plan d'adressage évolutif
- Le VPC doit disposer d'un nom d'hôte DNS et d'un support de résolution DNS. Dans le cas contraire, les nœuds ne peuvent pas enregistrer le cluster client. Pour plus d'informations, consultez DNS attributes for your VPC (Attributs DNS pour votre VPC) dans le Guide de l'utilisateur d'Amazon VPC.
- Le VPC peut nécessiter l'utilisation de points de terminaison VPC AWS PrivateLink pour pouvoir contacter l'API PCS. AWS Pour plus d'informations, consultez Connecter votre VPC aux services à l'aide du guide de AWS PrivateLink l'utilisateur Amazon VPC.

> **Important**
>
> AWS PCS ne prend pas en charge les VPC dotés d'une location d'instance dédiée. Le VPC que vous utilisez pour les AWS PCS doit utiliser la location d'`default`instance. Vous pouvez modifier la location de l'instance pour un VPC existant. Pour plus d'informations, consultez Modifier la location d'instance d'un VPC dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Exigences et considérations requises pour les sous-réseaux

Lorsque vous créez un cluster Slurm, AWS PCS crée une interface réseau élastique (ENI) dans le sous-réseau que vous avez spécifié. Cette interface réseau permet la communication entre le contrôleur du planificateur et le VPC du client. L'interface réseau permet également à Slurm de communiquer avec les composants déployés dans votre compte. Vous ne pouvez spécifier le sous-réseau d'un cluster qu'au moment de sa création.

Exigences requises pour les sous-réseaux des clusters

Le sous-réseau que vous spécifiez lors de la création d'un cluster doit répondre aux exigences suivantes :

- Le sous-réseau doit avoir au moins une adresse IP pour être utilisé par AWS PCS.
- Si votre cluster l'utilise IPv6, tous les sous-réseaux de votre cluster doivent l'utiliser IPv6.

> **Important**
>
> Les groupes de nœuds de calcul configurés avec un échantillon AWS PCS AMIs et plusieurs interfaces réseau ne fonctionneront pas actuellement si les sous-réseaux sont uniquement configurés pour être utilisés IPv6. Utilisez plutôt des sous-réseaux à double pile (IPv4 et IPv6) ou des sous-réseaux IPv4 uniquement. Pour de plus amples informations, veuillez consulter Utilisation d'exemples d'Amazon Machine Images (AMIs) avec AWS PCS.

- Le sous-réseau ne peut pas résider dans AWS Outposts une zone AWS locale ou dans une telle zone. AWS Wavelength
- Le sous-réseau peut être public ou privé. Nous vous recommandons de spécifier un sous-réseau privé, si possible. Un sous-réseau public est un sous-réseau avec une table de routage qui inclut une route vers une passerelle Internet ; un sous-réseau privé est un sous-réseau avec une table de routage qui n'inclut pas de route vers une passerelle Internet.

Exigences requises pour les sous-réseaux des nœuds

Vous pouvez déployer des nœuds et d'autres ressources de cluster sur le sous-réseau que vous spécifiez lors de la création de votre cluster AWS PCS, ainsi que sur d'autres sous-réseaux du même VPC.

Tout sous-réseau sur lequel vous déployez des nœuds et des ressources de cluster doit répondre aux exigences suivantes :

- Vous devez vous assurer que le sous-réseau dispose de suffisamment d'adresses IP disponibles pour déployer tous les nœuds et les ressources du cluster.
- Si votre cluster utilise IPv4 et que vous prévoyez de déployer des nœuds sur un sous-réseau public, ce sous-réseau doit attribuer automatiquement IPv4 des adresses publiques.

> **Note**
>
> Les instances d'un sous-réseau public doivent utiliser un groupe de sécurité avec des règles entrantes qui autorisent le trafic provenant d'adresses IP publiques. À moins que vous n'ayez des restrictions d'adresse source spécifiques, cela signifie une adresse IPv4 source de 0.0.0.0/0 ou une adresse IPv6 source de : :/0.

- Si le sous-réseau sur lequel vous déployez des nœuds est un sous-réseau privé et que sa table de routage n'inclut pas de route vers un périphérique de traduction d'adresses réseau (NAT) (IPv4), ajoutez des points de terminaison VPC à AWS PrivateLink l'aide du VPC du client. Les points de terminaison VPC sont nécessaires pour tous les AWS services contactés par les nœuds. Le seul point de terminaison requis est que le AWS PCS autorise le nœud à appeler l'action de l'`RegisterComputeNodeGroupInstance`API. Pour plus d'informations, consultez le RegisterComputeNodeGroupInstancemanuel de référence de l'API AWS PCS.
- L'état du sous-réseau public ou privé n'a aucun impact sur les AWS PCS ; les points de terminaison requis doivent être accessibles.

Création d'un VPC pour votre AWS cluster PCS

Vous pouvez créer un Amazon Virtual Private Cloud (Amazon VPC) pour vos clusters au sein de AWS Parallel Computing Service (AWS PCS).

Utilisez Amazon VPC pour lancer des ressources VPC dans un réseau virtuel que vous avez défini. Ce réseau virtuel ressemble beaucoup à un réseau traditionnel que vous pourriez exécuter dans votre propre centre de données. Toutefois, il présente l'avantage d'utiliser l'infrastructure évolutive d'Amazon Web Services. Nous vous recommandons de bien comprendre le service Amazon VPC avant de déployer des clusters VPC de production. Pour plus d'informations, consultez Qu'est-ce qu'Amazon VPC ? dans le mode visuel de l'auteur. Guide de l'utilisateur d'Amazon VPC.

Un cluster PCS, des nœuds et des ressources de support (telles que des systèmes de fichiers et des services d'annuaire) sont déployés au sein de votre Amazon VPC. Si vous souhaitez utiliser un Amazon VPC existant avec PCS, celui-ci doit répondre aux exigences décrites dans. AWS Exigences et considérations relatives au PCS, au VPC et aux sous-réseaux Cette rubrique explique comment créer un VPC répondant aux exigences du PCS à l'aide d'un modèle fourni AWS. CloudFormation Une fois que vous avez déployé un modèle, vous pouvez consulter les ressources créées par le modèle pour savoir exactement quelles ressources il a créées, et la configuration de ces ressources.

Conditions préalables

Pour créer un Amazon VPC pour PCS, vous devez disposer des autorisations IAM nécessaires pour créer des ressources Amazon VPC. Ces ressources sont les sous-réseaux VPCs, les groupes de sécurité, les tables de routage et les routes, ainsi que les passerelles Internet et NAT. Pour plus d'informations, consultez la section Créer un VPC avec un sous-réseau public dans le guide de l'utilisateur Amazon VPC. Pour consulter la liste complète d'Amazon EC2, consultez Actions, ressources et clés de condition pour Amazon EC2 dans le Service Authorization Reference.

Création d'un Amazon VPC

Créez un VPC en copiant et en collant l'URL appropriée pour l' Région AWS endroit où vous utiliserez PCS. Vous pouvez également télécharger le CloudFormation modèle et le charger vous-même sur la CloudFormation console.

- USA Est (Virginie du Nord) (us-east-1)

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

- USA Est (Ohio) (us-east-2)

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

- USA Ouest (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
```

- Modèle uniquement

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

Pour créer un Amazon VPC pour PCS

1. Ouvrez le modèle dans la CloudFormation console.

   > **Note**
   >
   > Elles sont préremplies dans le modèle afin que vous puissiez simplement les laisser comme valeurs par défaut.

2. Sous Fournir un nom de pile, puis Nom de pile, entrez`hpc-networking`.
3. Dans la section Paramètres, entrez les informations suivantes :

   a. Sous VPC, entrez ensuite CidrBlock`10.3.0.0/16`

   b. Sous les sous-réseaux A :

      i. Puis CidrPublicSubnetA, entrez `10.3.0.0/20`

      ii. Puis CidrPrivateSubnetA, entrez `10.3.128.0/20`

   c. Sous les sous-réseaux B :

      i. Puis CidrPublicSubnetB, entrez `10.3.16.0/20`

      ii. Puis CidrPrivateSubnetA, entrez `10.3.144.0/20`

   d. Sous les sous-réseaux C :

      i. Pour ProvisionSubnetsC, sélectionnez`True`.

      > **Note**
      >
      > Si vous créez un VPC dans une région comportant moins de trois zones de disponibilité, cette option sera ignorée si elle est définie sur. `True`

      ii. Puis CidrPublicSubnetB, entrez `10.3.32.0/20`

      iii. Puis CidrPrivateSubnetA, entrez `10.3.160.0/20`

4. Sous Fonctionnalités, cochez la case Je reconnais qu'AWS CloudFormation peut créer des ressources IAM.

Surveillez l'état de la CloudFormation pile. Lorsqu'elle est `CREATE_COMPLETE` atteinte, la ressource VPC est prête à être utilisée.

> **Note**
>
> Pour voir toutes les ressources créées par le CloudFormation modèle, ouvrez la CloudFormation console. Choisissez la pile `hpc-networking`, puis choisissez l'onglet Ressources.

Groupes de sécurité dans AWS PCS

Les groupes de sécurité d'Amazon EC2 agissent comme des pare-feux virtuels pour contrôler le trafic entrant et sortant vers les instances. Utilisez un modèle de lancement pour un groupe de nœuds de calcul AWS PCS afin d'ajouter ou de supprimer des groupes de sécurité à ses instances. Si votre modèle de lancement ne contient aucune interface réseau, utilisez-le `SecurityGroupIds` pour fournir une liste de groupes de sécurité. Si votre modèle de lancement définit des interfaces réseau, vous devez utiliser le `Groups` paramètre pour attribuer des groupes de sécurité à chaque interface réseau. Pour en savoir plus sur l'utilisation des modèles de lancement, consultez Utilisation des modèles de lancement Amazon EC2 avec PCS AWS.

> **Note**
>
> Les modifications apportées à la configuration du groupe de sécurité dans le modèle de lancement concernent uniquement les nouvelles instances lancées après la mise à jour du groupe de nœuds de calcul.

Exigences et considérations relatives aux groupes de sécurité

AWS PCS crée une interface réseau élastique (ENI) entre comptes dans le sous-réseau que vous spécifiez lors de la création d'un cluster. Cela fournit au planificateur HPC, qui s'exécute dans un compte géré par AWS, un chemin pour communiquer avec les instances EC2 lancées par PCS.

AWS Vous devez fournir un groupe de sécurité pour cette ENI qui autorise la communication bidirectionnelle entre le planificateur ENI et les instances EC2 de votre cluster.

Un moyen simple d'y parvenir consiste à créer un groupe de sécurité autoréférencé permissif qui autorise le TCP/IP trafic sur tous les ports entre tous les membres du groupe. Vous pouvez l'associer à la fois au cluster et aux instances EC2 du groupe de nœuds.

Exemple de configuration de groupe de sécurité permissive

IPv4

Type de règle	Protocoles	Ports	Source	Destination
Entrant	Tous	Tous	Self	
Sortant	Tous	Tous		0.0.0.0/0
Sortant	Tous	Tous		Self

IPv6

Type de règle	Protocoles	Ports	Source	Destination
Entrant	Tous	Tous	Self	
Sortant	Tous	Tous		::/0
Sortant	Tous	Tous		Self

Ces règles permettent à tout le trafic de circuler librement entre le contrôleur Slurm et les nœuds, autorisent tout le trafic sortant vers n'importe quelle destination et activent le trafic EFA.

Exemple de configuration restrictive d'un groupe de sécurité

Vous pouvez également limiter les ports ouverts entre le cluster et ses nœuds de calcul. Pour le planificateur Slurm, le groupe de sécurité rattaché à votre cluster doit autoriser les ports suivants :

- 6817 — activer les connexions entrantes `slurmctld` depuis des instances EC2

- 6818 — active les connexions sortantes depuis et en cours d'`slurmd`exécution sur `slurmctld` les instances EC2

Le groupe de sécurité attaché à vos nœuds de calcul doit autoriser les ports suivants :

- 6817 — active les connexions sortantes `slurmctld` depuis des instances EC2.
- 6818 — activer les connexions entrantes et sortantes vers `slurmctld` et `slurmd` depuis des instances de groupes `slurmd` de nœuds
- 60001—63000 — connexions entrantes et sortantes entre les instances de groupes de nœuds à prendre en charge `srun`
- Trafic EFA entre les instances de groupes de nœuds. Pour plus d'informations, voir Préparer un groupe de sécurité compatible EFA dans le Guide de l'utilisateur pour les instances Linux
- Tout autre trafic inter-nœuds requis par votre charge de travail

Plusieurs interfaces réseau dans les AWS PCS

Certaines instances EC2 possèdent plusieurs cartes réseau. Cela leur permet de fournir des performances réseau supérieures, notamment des capacités de bande passante supérieures à 100 Gbit/s et une meilleure gestion des paquets. Pour plus d'informations sur les instances dotées de plusieurs cartes réseau, consultez les interfaces réseau Elastic dans le guide de l'utilisateur Amazon Elastic Compute Cloud.

Configurez des cartes réseau supplémentaires pour les instances d'un groupe de nœuds de calcul AWS PCS en ajoutant des interfaces réseau à son modèle de lancement EC2. Vous trouverez ci-dessous un exemple de modèle de lancement qui active deux cartes réseau, comme celles que l'on trouve sur une `hpc7a.96xlarge` instance. Prenez note des détails suivants :

- Le sous-réseau de chaque interface réseau doit être le même que celui que vous avez choisi lors de la configuration du groupe de nœuds de calcul AWS PCS qui utilisera le modèle de lancement.
- Le périphérique réseau principal, sur lequel les communications réseau de routine telles que le trafic SSH et HTTPS auront lieu, est établi en définissant un `DeviceIndex` de`0`. Les autres interfaces réseau ont un `DeviceIndex` de`1`. Il ne peut y avoir qu'une seule interface réseau principale ; toutes les autres interfaces sont secondaires.
- Toutes les interfaces réseau doivent avoir un identifiant unique`NetworkCardIndex`. Il est recommandé de les numéroter de manière séquentielle selon leur définition dans le modèle de lancement.

- Les groupes de sécurité pour chaque interface réseau sont définis à l'aide de`Groups`. Dans cet exemple, un groupe de sécurité SSH entrant (`sg-`*`SshSecurityGroupId`*) est ajouté à l'interface réseau principale, ainsi que le groupe de sécurité permettant les communications au sein du cluster (). `sg-`*`ClusterSecurityGroupId`* Enfin, un groupe de sécurité autorisant les connexions sortantes vers Internet (`sg-`*`InternetOutboundSecurityGroupId`*) est ajouté aux interfaces principale et secondaire.

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

Groupes de placement pour les instances EC2 dans PCS AWS

Vous pouvez utiliser un groupe de placement pour influencer le placement des instances EC2 afin de répondre aux besoins de la charge de travail qui les exécute.

Types de groupes de placement

- Cluster — Regroupe les instances les unes aux autres dans une zone de disponibilité afin d'optimiser les communications à faible latence.
- Partition : répartit les instances sur des partitions logiques pour optimiser la résilience.

- Spread : impose strictement le lancement d'un petit nombre d'instances sur du matériel distinct, ce qui peut également contribuer à la résilience.

Pour plus d'informations, consultez la section Groupes de placement pour vos instances Amazon EC2 dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Nous vous recommandons d'inclure un groupe de placement de clusters lorsque vous configurez un groupe de nœuds de calcul AWS PCS pour utiliser Elastic Fabric Adapter (EFA).

Pour créer un groupe de placement de clusters compatible avec EFA

1. Créez un groupe de placement avec le type cluster pour le groupe de nœuds de calcul.
   - Utilisez la AWS CLI commande suivante :

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

   - Vous pouvez également utiliser un CloudFormation modèle pour créer un groupe de placement. Pour plus d'informations, consultez la section Utilisation des CloudFormation modèles dans le Guide de AWS CloudFormation l'utilisateur. Téléchargez le modèle à partir de l'URL suivante et chargez-le dans la CloudFormation console.

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

2. Incluez le groupe de placement dans le modèle de lancement EC2 pour le groupe de nœuds de calcul AWS PCS.

Utilisation d'Elastic Fabric Adapter (EFA) avec PCS AWS

Elastic Fabric Adapter (EFA) est une interconnexion réseau avancée à hautes performances que vous pouvez connecter AWS à votre instance EC2 pour accélérer les applications de calcul haute performance (HPC) et d'apprentissage automatique. L'activation de vos applications exécutées sur un cluster AWS PCS avec EFA implique de configurer les instances du groupe de nœuds de calcul AWS PCS pour utiliser EFA comme suit.

> **Note**
>
> Installer EFA sur une AWS AMI compatible avec le PC — Le pilote EFA doit être installé et chargé sur l'AMI utilisée dans le groupe de nœuds de calcul AWS PCS. Pour plus

d'informations sur la création d'une AMI personnalisée avec le logiciel EFA installé, consultezImages Amazon Machine personnalisées (AMIs) pour AWS PC.

Table des matières

- Identifier les instances EC2 compatibles avec EFA
- Création d'un groupe de sécurité pour prendre en charge les communications EFA
- (Facultatif) Créez un groupe de placement
- Création ou mise à jour d'un modèle de lancement EC2
- Création ou mise à jour de groupes de nœuds de calcul pour EFA
- (Facultatif) Testez EFA
- (Facultatif) Utilisez un CloudFormation modèle pour créer un modèle de lancement compatible avec l'EFA

Identifier les instances EC2 compatibles avec EFA

Pour utiliser EFA, tous les types d'instances autorisés pour un groupe de calcul AWS PCS doivent prendre en charge EFA et avoir le même nombre de v CPUs (et le cas GPUs échéant). Pour obtenir la liste des instances compatibles EFA, consultez Elastic Fabric Adapter pour les charges de travail HPC et ML sur Amazon EC2 dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud. Vous pouvez également utiliser le AWS CLI pour afficher la liste des types d'instances qui prennent en charge l'EFA. *region-code*Remplacez-le par l' Région AWS endroit où vous utilisez le AWS PCS, tel que`us-east-1`.

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

> **Note**
>
> Déterminez le nombre d'interfaces réseau disponibles : certaines instances EC2 possèdent plusieurs cartes réseau. Cela leur permet d'en avoir plusieurs EFAs. Pour de plus amples informations, veuillez consulter Plusieurs interfaces réseau dans les AWS PCS.

Création d'un groupe de sécurité pour prendre en charge les communications EFA

AWS CLI

Vous pouvez utiliser la AWS CLI commande suivante pour créer un groupe de sécurité prenant en charge l'EFA. La commande génère un ID de groupe de sécurité. Procédez aux remplacements suivants :

- *region-code*— Spécifiez l' Région AWS endroit où vous utilisez le AWS PCS, par exemple`us-east-1`.
- *vpc-id*— Spécifiez l'ID du VPC que vous utilisez pour AWS les PCS.
- *efa-group-name*— Indiquez le nom que vous avez choisi pour le groupe de sécurité.

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

Utilisez les commandes suivantes pour associer des règles de groupe de sécurité entrantes et sortantes. Effectuez le remplacement suivant :

- *efa-secgroup-id*— Indiquez l'ID du groupe de sécurité EFA que vous venez de créer.

```
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

Vous pouvez utiliser un CloudFormation modèle pour créer un groupe de sécurité prenant en charge l'EFA. Téléchargez le modèle à partir de l'URL suivante, puis chargez-le dans la AWS CloudFormation console.

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

Le modèle étant ouvert dans la AWS CloudFormation console, entrez les options suivantes.

- Sous Fournir un nom de pile
  - Sous Nom de la pile, entrez un nom tel que `efa-sg-stack`.
- Sous Paramètres
  - Sous SecurityGroupName, entrez un nom tel que `efa-sg`.
  - Sous VPC, sélectionnez le VPC dans lequel vous utiliserez PCS. AWS

Terminez la création de la CloudFormation pile et surveillez son état. Lorsqu'il atteint `CREATE_COMPLETE` l'EFA, le groupe de sécurité est prêt à être utilisé.

(Facultatif) Créez un groupe de placement

Nous vous recommandons de lancer toutes les instances qui utilisent EFA dans un groupe de placement de clusters afin de minimiser la distance physique entre elles. Créez un groupe de placement pour chaque groupe de nœuds de calcul dans lequel vous prévoyez d'utiliser EFA. Consultez Groupes de placement pour les instances EC2 dans PCS AWS la section pour créer un groupe de placement pour votre groupe de nœuds de calcul.

Création ou mise à jour d'un modèle de lancement EC2

Les interfaces réseau EFA sont configurées dans le modèle de lancement EC2 pour un groupe de nœuds de calcul AWS PCS. S'il existe plusieurs cartes réseau, plusieurs EFAs peuvent être configurées. Le groupe de sécurité EFA et le groupe de placement optionnel sont également inclus dans le modèle de lancement.

Voici un exemple de modèle de lancement pour les instances dotées de deux cartes réseau, telles que hpc7a.96xlarge. Les instances seront lancées `subnet-`*`SubnetID1`* dans un groupe de placement en clusterpg-*`PlacementGroupId1`*.

Les groupes de sécurité doivent être ajoutés spécifiquement à chaque interface EFA. Chaque EFA a besoin du groupe de sécurité qui active le trafic EFA (`sg-`*`EfaSecGroupId`*). Les autres groupes de sécurité, en particulier ceux qui gèrent le trafic normal comme SSH ou HTTPS, doivent uniquement être attachés à l'interface réseau principale (désignée par un `DeviceIndex` de`0`). Les modèles de lancement dans lesquels des interfaces réseau sont définies ne permettent pas de définir des groupes de sécurité à l'aide du `SecurityGroupIds` paramètre. Vous devez définir une valeur pour `Groups` chaque interface réseau que vous configurez.

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

Création ou mise à jour de groupes de nœuds de calcul pour EFA

Les groupes de nœuds de calcul de votre AWS PCS doivent contenir des instances présentant le même nombre de vCPUs, la même architecture de processeur et le même support EFA. Configurez

le groupe de nœuds de calcul pour utiliser l'AMI sur laquelle le logiciel EFA est installé et pour utiliser le modèle de lancement qui configure les interfaces réseau compatibles EFA.

(Facultatif) Testez EFA

Vous pouvez démontrer la communication compatible EFA entre deux nœuds d'un groupe de nœuds de calcul en exécutant le `fi_pingpong` programme, qui est inclus dans l'installation du logiciel EFA. Si ce test est réussi, il est probable que l'EFA soit correctement configuré.

Pour commencer, vous avez besoin de deux instances actives dans le groupe de nœuds de calcul. Si votre groupe de nœuds de calcul utilise une capacité statique, des instances devraient déjà être disponibles. Pour un groupe de nœuds de calcul utilisant la capacité dynamique, vous pouvez lancer deux nœuds à l'aide de la `salloc` commande. Voici un exemple d'un cluster avec un groupe de nœuds dynamique nommé hpc7g associé à une file d'attente nommée`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
```

Découvrez l'adresse IP des deux nœuds alloués à l'aide de`scontrol`. Dans l'exemple suivant, les adresses sont `10.3.140.69` pour hpc7g-1 et `10.3.132.211` pour`hpc7g-2`.

```
% scontrol show nodes hpc7g-[1-2]
NodeName=hpc7g-1 Arch=aarch64 CoresPerSocket=1
   CPUAlloc=0 CPUEfctv=64 CPUTot=64 CPULoad=0.00
   AvailableFeatures=hpc7g
   ActiveFeatures=hpc7g
   Gres=(null)
   NodeAddr=10.3.140.69 NodeHostName=ip-10-3-140-69 Version=25.05.4
   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=25.05.4
   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
```

Connectez-vous à l'un des nœuds (dans ce cas,`hpc7g-1`) à l'aide de SSH (ou SSM). Notez qu'il s'agit d'une adresse IP interne, vous devrez peut-être vous connecter depuis l'un de vos nœuds de connexion si vous utilisez SSH. Sachez également que l'instance doit être configurée avec une clé SSH via le modèle de lancement du groupe de nœuds de calcul.

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

Maintenant, `fi_pingpong` lancez-vous en mode serveur.

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

Connectez-vous à la deuxième instance (`hpc7g-2`).

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

Exécuter `fi_pingpong` en mode client, en se connectant au serveur sur`hpc7g-1`. Vous devriez voir une sortie similaire à l'exemple ci-dessous.

```
% /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
```

(Facultatif) Utilisez un CloudFormation modèle pour créer un modèle de lancement compatible avec l'EFA

Étant donné que la configuration d'EFA comporte plusieurs dépendances, un CloudFormation modèle a été fourni que vous pouvez utiliser pour configurer un groupe de nœuds de calcul. Il prend en charge les instances dotées d'un maximum de quatre cartes réseau. Pour en savoir plus sur les instances dotées de plusieurs cartes réseau, consultez les interfaces réseau élastiques dans le guide de l'utilisateur Amazon Elastic Compute Cloud.

Téléchargez le CloudFormation modèle à partir de l'URL suivante, puis chargez-le sur la CloudFormation console dans Région AWS laquelle vous utilisez AWS PCS.

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

Le modèle étant ouvert dans la CloudFormation console, entrez les valeurs suivantes. Notez que le modèle fournit certaines valeurs de paramètres par défaut. Vous pouvez les conserver comme valeurs par défaut.

- Sous Fournir un nom de pile
  - Sous Nom de la pile, entrez un nom descriptif. Nous vous recommandons d'incorporer le nom que vous choisirez pour votre groupe de nœuds de calcul AWS PCS, tel que*NODEGROUPNAME*`-efa-lt`.
- Sous Paramètres
  - Sous NumberOfNetworkCards, choisissez le nombre de cartes réseau dans les instances qui figureront dans votre groupe de nœuds.
  - Sous VpcId, choisissez le VPC sur lequel votre cluster AWS PCS est déployé.

- Sous NodeGroupSubnetId, choisissez le sous-réseau de votre VPC de cluster où les instances compatibles EFA seront lancées.
- Sous PlacementGroupName, laissez le champ vide pour créer un nouveau groupe de placement de clusters pour le groupe de nœuds. Si vous souhaitez utiliser un groupe de placement existant, entrez son nom ici.
- Sous ClusterSecurityGroupId, choisissez le groupe de sécurité que vous utilisez pour autoriser l'accès aux autres instances du cluster et à l'API AWS PCS. De nombreux clients choisissent le groupe de sécurité par défaut à partir de leur VPC de cluster.
- Sous SshSecurityGroupId, indiquez l'ID d'un groupe de sécurité que vous utilisez pour autoriser l'accès SSH entrant aux nœuds de votre cluster.
- Pour SshKeyName, sélectionnez la paire de clés SSH pour accéder aux nœuds de votre cluster.
- Pour LaunchTemplateName, entrez un nom descriptif pour le modèle de lancement, tel que*NODEGROUPNAME*`-efa-lt`. Le nom doit être unique à celui de Compte AWS l' Région AWS endroit où vous utiliserez le AWS PCS.

- Sous Capacités
  - Cochez la case Je reconnais que cela AWS CloudFormation pourrait créer des ressources IAM.

Surveillez l'état de la CloudFormation pile. Lorsqu'il atteint, `CREATE_COMPLETE` le modèle de lancement est prêt à être utilisé. Utilisez-le avec un groupe de nœuds de calcul AWS PCS, comme décrit ci-dessus dansCréation ou mise à jour de groupes de nœuds de calcul pour EFA.

Utilisation de systèmes de fichiers réseau avec AWS PCS

Vous pouvez associer des systèmes de fichiers réseau à des nœuds lancés dans un groupe de nœuds de calcul AWS PCS (AWS Parallel Computing Service) afin de fournir un emplacement permanent où les données et les fichiers peuvent être écrits et accessibles. Vous pouvez utiliser les systèmes de fichiers fournis par les AWS services, notamment Amazon Elastic File System (Amazon EFS), Amazon FSx for Lustre, Amazon FSx pour NetApp ONTAP, Amazon FSx pour OpenZFS et Amazon File Cache. Vous pouvez également utiliser des systèmes de fichiers autogérés, tels que des serveurs NFS.

Cette rubrique présente des considérations et des exemples relatifs à l'utilisation de systèmes de fichiers réseau avec AWS PCS.

Considérations relatives à l'utilisation de systèmes de fichiers réseau

Les détails de mise en œuvre des différents systèmes de fichiers sont différents, mais il existe des considérations communes.

- Le logiciel du système de fichiers approprié doit être installé sur l'instance. Par exemple, pour utiliser Amazon FSx pour Lustre, le Lustre package approprié doit être présent. Cela peut être accompli en l'incluant dans l'AMI du groupe de nœuds de calcul ou en utilisant un script qui s'exécute au démarrage de l'instance.
- Il doit exister une route réseau entre le système de fichiers réseau partagé et les instances du groupe de nœuds de calcul.
- Les règles du groupe de sécurité pour le système de fichiers réseau partagé et les instances du groupe de nœuds de calcul doivent autoriser les connexions aux ports concernés.
- Vous devez maintenir un espace de noms POSIX d'utilisateur et de groupe cohérent entre les ressources qui accèdent aux systèmes de fichiers. Dans le cas contraire, les tâches et les processus interactifs exécutés sur votre cluster PCS risquent de rencontrer des erreurs d'autorisation.
- Les montages de systèmes de fichiers sont effectués à l'aide de modèles de EC2 lancement. Des erreurs ou des délais d'attente lors du montage d'un système de fichiers réseau peuvent empêcher les instances d'être disponibles pour exécuter des tâches. Ceci, à son tour, peut entraîner

des coûts imprévus. Pour plus d'informations sur le débogage des modèles de lancement, consultezUtilisation des modèles de lancement Amazon EC2 avec PCS AWS.

Exemples de montages réseau

Vous pouvez créer des systèmes de fichiers à l'aide d'Amazon EFS, Amazon FSx for Lustre, Amazon FSx pour NetApp ONTAP, Amazon FSx pour OpenZFS et Amazon File Cache. Développez la section correspondante ci-dessous pour voir un exemple de chaque montage réseau.

Amazon EFS

Configuration du système de fichiers

Créez un système de fichiers Amazon EFS. Assurez-vous qu'il dispose d'une cible de montage dans chaque zone de disponibilité où vous lancerez des instances de groupes de nœuds de calcul PCS. Assurez-vous également que chaque cible de montage est associée à un groupe de sécurité qui autorise l'accès entrant et sortant depuis les instances du groupe de nœuds de calcul PCS. Pour plus d'informations, consultez la section Mount targets and security groups dans le manuel Amazon Elastic File System User Guide.

Modèle de lancement

Ajoutez le ou les groupes de sécurité de la configuration de votre système de fichiers au modèle de lancement que vous utiliserez pour le groupe de nœuds de calcul.

Incluez les données utilisateur qui utilisent le `cloud-config` mécanisme de montage du système de fichiers Amazon EFS. Remplacez les valeurs suivantes dans ce script par vos propres informations :

- *`mount-point-directory`*— Le chemin de chaque instance sur laquelle vous allez monter Amazon EFS
- *`filesystem-id`*— L'ID du système de fichiers EFS

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

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

packages:
```

```
  - amazon-efs-utils

runcmd:
  - mkdir -p /mount-point-directory
  - echo "filesystem-id:/ /mount-point-directory efs tls,_netdev" >> /etc/fstab
  - mount -a -t efs defaults

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

Amazon FSx pour Lustre

Configuration du système de fichiers

Créez un système de fichiers FSx pour Lustre dans le VPC où vous AWS utiliserez PCS. Pour minimiser les transferts entre zones, déployez dans un sous-réseau de la même zone de disponibilité où vous lancerez la majorité de vos instances de groupes de nœuds de calcul PCS. Assurez-vous que le système de fichiers est associé à un groupe de sécurité qui autorise l'accès entrant et sortant depuis les instances du groupe de nœuds de calcul PCS. Pour plus d'informations sur les groupes de sécurité, consultez la section Contrôle d'accès au système de fichiers avec Amazon VPC dans le guide de l'utilisateur Amazon FSx for Lustre.

Modèle de lancement

Incluez les données utilisateur utilisées `cloud-config` pour monter le système de fichiers FSx for Lustre. Remplacez les valeurs suivantes dans ce script par vos propres informations :

- *`mount-point-directory`*— Le chemin d'une instance sur laquelle vous souhaitez effectuer le montage FSx pour Lustre
- *`filesystem-id`*— L'ID du système de fichiers pour le système de fichiers FSx for Lustre
- *`mount-name`*— Le nom de montage du système de fichiers FSx for Lustre
- *`region-code`*— L' Région AWS endroit où le système de fichiers FSx for Lustre est déployé (doit être le même que celui de votre système AWS PCS)
- (Facultatif) *`latest`* — N'importe quelle version de FSx for Lustre Lustre prise en charge par

```
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 pour NetApp ONTAP

Configuration du système de fichiers

Créez un système de fichiers Amazon FSx for NetApp ONTAP dans le VPC où vous utiliserez AWS PCS. Pour minimiser les transferts entre zones, déployez dans un sous-réseau de la même zone de disponibilité où vous lancerez la majorité de vos instances de groupes de nœuds de calcul AWS PCS. Assurez-vous que le système de fichiers est associé à un groupe de sécurité qui autorise l'accès entrant et sortant depuis les instances du groupe de nœuds de calcul AWS PCS. Pour plus d'informations sur les groupes de sécurité, consultez la section Contrôle d'accès au système de fichiers avec Amazon VPC dans le guide de l'utilisateur FSx pour ONTAP.

Modèle de lancement

Incluez les données utilisateur utilisées `cloud-config` pour monter le volume racine d'un système de fichiers FSx pour ONTAP. Remplacez les valeurs suivantes dans ce script par vos propres informations :

- *`mount-point-directory`*— Le chemin d'une instance sur laquelle vous souhaitez monter votre volume FSx for ONTAP
- *`svm-id`*— L'identifiant de la SVM pour le système de fichiers FSx for ONTAP
- *`filesystem-id`*— L'ID du système de fichiers pour le système de fichiers FSx for ONTAP
- *`region-code`*— L' Région AWS endroit où le système de fichiers FSx for ONTAP est déployé (doit être le même que celui de votre système AWS PCS)
- *`volume-name`*— Le nom du volume FSx for ONTAP

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

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

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

--==MYBOUNDARY==
```

Amazon FSx pour OpenZFS

Configuration du système de fichiers

Créez un système de fichiers FSx pour OpenZFS dans le VPC où vous utiliserez PCS. AWS Pour minimiser les transferts entre zones, déployez dans un sous-réseau de la même zone de disponibilité où vous lancerez la majorité de vos instances de groupes de nœuds de calcul AWS PCS. Assurez-vous que le système de fichiers est associé à un groupe de sécurité qui autorise l'accès entrant et sortant depuis les instances du groupe de nœuds de calcul AWS PCS. Pour plus d'informations sur les groupes de sécurité, consultez la section Gestion de l'accès au système de fichiers avec Amazon VPC dans le guide de l'utilisateur FSx d'OpenZFS.

Modèle de lancement

Incluez les données utilisateur utilisées `cloud-config` pour monter le volume racine d'un système FSx de fichiers OpenZFS. Remplacez les valeurs suivantes dans ce script par vos propres informations :

- *`mount-point-directory`*— Le chemin d'une instance sur laquelle vous souhaitez monter votre partage FSx pour OpenZFS
- *`filesystem-id`*— L'ID du système de fichiers FSx pour le système de fichiers OpenZFS
- *`region-code`*— L' Région AWS endroit où le système FSx de fichiers OpenZFS est déployé (doit être le même que celui de votre système AWS PCS)

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

--==MYBOUNDARY==
```

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

runcmd:
- mkdir -p /mount-point-directory
- mount -t nfs -o noatime,nfsvers=4.2,sync,rsize=1048576,wsize=1048576 filesystem-
id.fsx.region-code.amazonaws.com:/fsx/ /mount-point-directory

--==MYBOUNDARY==
```

Cache de fichiers Amazon

Configuration du système de fichiers

Créez un cache de fichiers Amazon dans le VPC où vous AWS utiliserez PCS. Pour minimiser les transferts entre zones, choisissez un sous-réseau dans la même zone de disponibilité où vous lancerez la majorité de vos instances de groupes de nœuds de calcul PCS. Assurez-vous que le cache de fichiers est associé à un groupe de sécurité qui autorise le trafic entrant et sortant sur le port 988 entre vos instances PCS et le cache de fichiers. Pour plus d'informations sur les groupes de sécurité, consultez la section Contrôle d'accès au cache avec Amazon VPC dans le guide de l'utilisateur Amazon File Cache.

Modèle de lancement

Ajoutez le ou les groupes de sécurité de la configuration de votre système de fichiers au modèle de lancement que vous utiliserez pour le groupe de nœuds de calcul.

Incluez les données utilisateur utilisées `cloud-config` pour monter l'Amazon File Cache. Remplacez les valeurs suivantes dans ce script par vos propres informations :

- *`mount-point-directory`*— Le chemin d'une instance sur laquelle vous souhaitez effectuer le montage FSx pour Lustre
- *`cache-dns-name`*— Le nom du système de noms de domaine (DNS) pour le cache de fichiers
- *`mount-name`*— Le nom de montage pour le cache de fichiers

```
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) pour AWS PC

AWS Le PCS fonctionne avec AMIs ce que vous proposez, offrant ainsi une grande flexibilité en termes de logiciel et de configuration sur les nœuds de votre cluster. Si vous essayez AWS PCS, vous pouvez utiliser un exemple d'AMI fourni et géré par AWS. Si vous utilisez des AWS PCS en production, nous vous recommandons de créer le vôtre AMIs. Cette rubrique explique comment découvrir et utiliser l'exemple AMIs, ainsi que comment créer et utiliser votre propre version personnalisée AMIs.

Rubriques

- Utilisation d'exemples d'Amazon Machine Images (AMIs) avec AWS PCS
- Images Amazon Machine personnalisées (AMIs) pour AWS PC
- Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS
- Notes de mise à jour pour un exemple de AWS PCS AMIs

Utilisation d'exemples d'Amazon Machine Images (AMIs) avec AWS PCS

AWS fournit un exemple AMIs que vous pouvez utiliser comme point de départ pour travailler avec AWS PCS.

> **Important**
>
> AMIs Les échantillons sont fournis à des fins de démonstration et ne sont pas recommandés pour les charges de travail de production.

> **Important**
>
> Les groupes de nœuds de calcul configurés avec un échantillon AWS PCS AMIs et plusieurs interfaces réseau ne fonctionneront pas actuellement si les sous-réseaux sont uniquement configurés pour être utilisés IPv6. Utilisez plutôt des sous-réseaux à double pile (IPv4 et IPv6) ou des sous-réseaux IPv4 uniquement.

Trouver un échantillon AWS PCS actuel AMIs

AWS Management Console

AMIs Les exemples d'AWS PCS ont la convention de dénomination suivante :

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

Valeurs acceptées

- *OS* – amzn2
- *architecture* – x86_64 ou arm64
- *scheduler* – slurm
- *scheduler-major-version* – 25.05

Pour trouver un échantillon AWS PCS AMIs

1. Ouvrez la EC2 console Amazon.
2. Accédez à AMIs.
3. Choisissez Images publiques.
4. Dans Rechercher une AMI par attribut ou balise, recherchez une AMI en utilisant le nom du modèle.

   Exemples

   - Exemple d'AMI pour Slurm 25.05 sur des instances Arm64

     ```
     aws-pcs-sample_ami-amzn2-arm64-slurm-25.05
     ```

   - Exemple d'AMI pour Slurm 25.05 sur des instances x86

     ```
     aws-pcs-sample_ami-amzn2-x86_64-slurm-25.05
     ```

   > **Note**
   >
   > S'il y en a plusieurs AMIs, utilisez l'AMI avec l'horodatage le plus récent.

5. Utilisez l'ID AMI lorsque vous créez ou mettez à jour un groupe de nœuds de calcul.

AWS CLI

Vous pouvez trouver le dernier exemple d'AMI AWS PCS avec les commandes suivantes. *region-code*Remplacez-le par l' Région AWS endroit où vous utilisez le AWS PCS, tel queus-east-1.

- x86_64

```
aws ec2 describe-images --region region-code --owners amazon \
--filters 'Name=name,Values=aws-pcs-sample_ami-amzn2-x86_64-slurm-25.05*' \
            'Name=state,Values=available' \
--query 'sort_by(Images, &CreationDate)[-1].[Name,ImageId]' --output text
```

- Bras 64

```
aws ec2 describe-images --region region-code --owners amazon \
--filters 'Name=name,Values=aws-pcs-sample_ami-amzn2-arm64-slurm-25.05*' \
            'Name=state,Values=available' \
--query 'sort_by(Images, &CreationDate)[-1].[Name,ImageId]' --output text
```

Utilisez l'ID AMI lorsque vous créez ou mettez à jour un groupe de nœuds de calcul.

En savoir plus sur l'échantillon AWS PCS AMIs

Pour consulter le contenu et les détails de configuration des versions actuelles et précédentes de l'exemple AWS PCS AMIs, voirNotes de mise à jour pour un exemple de AWS PCS AMIs.

Créez le vôtre AMIs compatible avec les AWS PCS

Pour savoir comment créer le vôtre compatible avec AMIs les AWS PCS, consultezImages Amazon Machine personnalisées (AMIs) pour AWS PC.

Images Amazon Machine personnalisées (AMIs) pour AWS PC

AWS PCS est conçu pour fonctionner avec les Amazon Machine Images (AMI) que vous apportez au service. Des logiciels et des configurations arbitraires AMIs peuvent être installés sur ceux-ci, à

condition que l'agent AWS PCS et une version compatible de Slurm soient installés et configurés correctement. Vous devez utiliser les programmes d'installation AWS fournis pour installer le logiciel AWS PCS sur votre AMI personnalisée. Nous vous recommandons d'utiliser les programmes d'installation AWS fournis pour installer Slurm sur votre AMI personnalisée, mais vous pouvez installer Slurm vous-même si vous préférez (ce n'est pas recommandé).

> **Note**
>
> Si vous souhaitez essayer AWS PCS sans créer d'AMI personnalisée, vous pouvez utiliser un exemple d'AMI fourni par AWS. Pour de plus amples informations, veuillez consulter Utilisation d'exemples d'Amazon Machine Images (AMIs) avec AWS PCS.

> **Important**
>
> AWS Le PCS nécessite actuellement un noyau prenant en IPv4 charge la communication entre nœuds locaux, même lorsque vous utilisez le AWS PCS dans un réseau IPv6 uniquement.

Ce didacticiel vous aide à créer une AMI qui peut être utilisée avec les groupes de nœuds de calcul PCS pour alimenter votre HPC et vos charges AI/ML de travail.

Rubriques

- Étape 1 — Lancer une instance temporaire
- Étape 2 — Installation de l'agent AWS PCS
- Étape 3 — Installation de Slurm
- Étape 4 — (Facultatif) Installation de pilotes, de bibliothèques et de logiciels d'application supplémentaires
- Étape 5 — Création d'une AMI compatible avec AWS PCS
- Étape 6 — Utiliser l'AMI personnalisée avec un groupe de nœuds de calcul AWS PCS
- Étape 7 — Mettre fin à l'instance temporaire

Étape 1 — Lancer une instance temporaire

Lancez une instance temporaire que vous pouvez utiliser pour installer et configurer le logiciel AWS PCS et le planificateur Slurm. Vous utilisez cette instance pour créer une AMI compatible avec AWS PCS.

Pour lancer une instance temporaire

1. Ouvrez la console Amazon EC2.
2. Dans le volet de navigation, choisissez Instances, puis choisissez Launch instances pour ouvrir le nouvel assistant de lancement d'instances.
3. (Facultatif) Dans la section Nom et balises, saisissez un nom pour l'instance, par exemple`PCS-AMI-instance`. Le nom est attribué à l'instance en tant qu'identification de ressource (`Name=PCS-AMI-instance`).
4. Dans la section Application and OS Images (Images d'applications et de systèmes d'exploitation), sélectionnez une AMI pour l'un des systèmes d'exploitation pris en charge.
5. Dans la section Instance type (Type d'instance), sélectionnez un type d'instance pris en charge.
6. Dans la section Key pair (Paire de clés), sélectionnez la paire de clés à utiliser pour l'instance.
7. Dans la section Paramètres réseau :
   - Pour Firewall (groupes de sécurité), choisissez Sélectionner un groupe de sécurité existant, puis sélectionnez un groupe de sécurité qui autorise l'accès SSH entrant à votre instance.
8. Dans la section Storage (Stockage), configurez les volumes selon vos besoins. Assurez-vous de configurer suffisamment d'espace pour installer vos propres applications et bibliothèques.
9. Dans le panneau Summary (Récapitulatif), sélectionnez Launch instance (Lancer l'instance).

Étape 2 — Installation de l'agent AWS PCS

Installez l'agent qui configure les instances lancées par AWS PCS pour une utilisation avec Slurm. Pour plus d'informations sur l'agent AWS PCS, consultezAWS Versions de l'agent PCS.

Pour installer l'agent AWS PCS

1. Connectez-vous à l'instance que vous avez lancée. Pour plus d'informations, consultez Connect to your Linux instance.

2. (Facultatif) Pour vous assurer que tous vos packages logiciels sont à jour, effectuez une mise à jour logicielle rapide sur votre instance. Ce processus peut prendre quelques minutes.

   - Amazon Linux 2, Amazon Linux 2023, RHEL 9, RHEL 8, Rocky Linux 9 et Rocky Linux 8

     ```
     sudo yum update -y
     ```

   - Ubuntu 22.04 et Ubuntu 24.04

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

3. Redémarrez l'instance et reconnectez-vous à celle-ci.
4. Téléchargez les fichiers d'installation de l'agent AWS PCS. Les fichiers d'installation sont regroupés dans un fichier tarball (`.tar.gz`) compressé. Pour télécharger la version stable la plus récente, utilisez la commande suivante. Remplacez *`region`* par l' Région AWS endroit où vous avez lancé votre instance temporaire, par exemple`us-east-1`.

   ```
   curl https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.3.2-1.tar.gz -o aws-pcs-agent-v1.3.2-1.tar.gz
   ```

   Vous pouvez également obtenir la dernière version en remplaçant le numéro de version par `latest` dans la commande précédente (par exemple :`aws-pcs-agent-v1-latest.tar.gz`).

   > **Note**
   >
   > Cela pourrait changer dans les futures versions du logiciel de l'agent AWS PCS.

5. (Facultatif) Vérifiez l'authenticité et l'intégrité de l'archive tar du logiciel AWS PCS. Nous vous recommandons de le faire pour vérifier l'identité de l'éditeur du logiciel et pour vérifier que le fichier n'a pas été modifié ou endommagé depuis sa publication.

   a. Téléchargez la clé GPG publique pour AWS PC et importez-la dans votre trousseau de clés. Remplacez *`region`* par l' Région AWS endroit où vous avez lancé votre instance temporaire. La commande doit renvoyer une valeur clé. Enregistrez la valeur clé ; vous l'utiliserez à l'étape suivante.

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

b. Exécutez la commande suivante pour vérifier l'empreinte digitale de la clé GPG.

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

La commande doit renvoyer une empreinte identique à ce qui suit :

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

> **Important**
>
> N'exécutez pas le script d'installation de l'agent AWS PCS si l'empreinte digitale ne correspond pas. Contactez AWS Support.

c. Téléchargez le fichier de signature et vérifiez la signature du fichier tar du logiciel AWS PCS. Remplacez *region* par l' Région AWS endroit où vous avez lancé votre instance temporaire, par exemple`us-east-1`.

```
wget https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.3.2-1.tar.gz.sig && \
    gpg --verify ./aws-pcs-agent-v1.3.2-1.tar.gz.sig
```

La sortie doit ressembler à ce qui suit :

```
gpg: assuming signed data in './aws-pcs-agent-v1.3.2-1.tar.gz'
gpg: Signature made Thu 06 Nov 2025 11:10:36 AM CET using RSA key ID ECC0AE5C
gpg: Good signature from "AWS PCS Packages (AWS PCS Packages)"
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 1C24 32C1 862F 64D1 F90A  239A 7EEF 030E DDF5 C21C
   Subkey fingerprint: B7E1 8788 3517 6A74 C3D5  EAF5 6088 136D ECC0 AE5C
```

Si le résultat inclut `Good signature` et que l'empreinte correspond à l'empreinte renvoyée à l'étape précédente, passez à l'étape suivante.

> **Important**
>
> N'exécutez pas le script d'installation du logiciel AWS PCS si l'empreinte digitale ne correspond pas. Contactez AWS Support.

6. Extrayez les fichiers du `.tar.gz` fichier compressé et accédez au répertoire extrait.

```
tar -xf aws-pcs-agent-v1.3.2-1.tar.gz && \
    cd aws-pcs-agent
```

7. Installez le logiciel AWS PCS.

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

8. Vérifiez le fichier de version du logiciel AWS PCS pour confirmer la réussite de l'installation.

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

La sortie doit ressembler à ce qui suit :

```
AGENT_INSTALL_DATE='Fri Dec 13 12:28:43 UTC 2024'
AGENT_VERSION='1.3.2'
AGENT_RELEASE='1'
```

Étape 3 — Installation de Slurm

Installez une version de Slurm compatible avec AWS PCS. Pour de plus amples informations, veuillez consulter Versions Slurm en PCS AWS.

> **Note**
>
> Si vous possédez une AMI sur laquelle une version précédente du logiciel Slurm est installée, vous devez effectuer les étapes suivantes pour installer la nouvelle version de Slurm. L'agent AWS PCS active la version correcte des fichiers binaires de Slurm au moment de l'exécution, conformément à la version de Slurm configurée au moment de la création du cluster.

Pour installer Slurm

1. Connectez-vous à la même instance temporaire sur laquelle vous avez installé le logiciel AWS PCS.
2. Téléchargez le logiciel d'installation Slurm. Le programme d'installation de Slurm est intégré dans un fichier tarball () `.tar.gz` compressé. Pour télécharger la version stable la plus récente, utilisez la commande suivante. *`region`*Remplacez-le par celui Région AWS de votre instance temporaire, tel que`us-east-1`.

   ```
   curl https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-25.05-installer-25.05.4-1.tar.gz \
        -o aws-pcs-slurm-25.05-installer-25.05.4-1.tar.gz
   ```

   Vous pouvez également obtenir la dernière version en remplaçant le numéro de version par `latest` dans la commande précédente (par exemple :`aws-pcs-slurm-25.05-installer-latest.tar.gz`). Pour une liste complète des versions disponibles avec des checksums, voirVersions Slurm en PCS AWS.

   > **Note**
   >
   > Cela pourrait changer dans les futures versions du logiciel d'installation Slurm.

3. (Facultatif) Vérifiez l'authenticité et l'intégrité de l'archive d'installation de Slurm. Nous vous recommandons de le faire pour vérifier l'identité de l'éditeur du logiciel et pour vérifier que le fichier n'a pas été modifié ou endommagé depuis sa publication.

   a. Téléchargez la clé GPG publique pour AWS PC et importez-la dans votre trousseau de clés. Remplacez *`region`* par l' Région AWS endroit où vous avez lancé votre instance temporaire. La commande doit renvoyer une valeur clé. Enregistrez la valeur clé ; vous l'utiliserez à l'étape suivante.

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

   b. Exécutez la commande suivante pour vérifier l'empreinte digitale de la clé GPG.

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

La commande doit renvoyer une empreinte identique à ce qui suit :

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

> **Important**
>
> N'exécutez pas le script d'installation de Slurm si l'empreinte digitale ne correspond pas. Contactez AWS Support.

c. Téléchargez le fichier de signature et vérifiez la signature du fichier tar du programme d'installation de Slurm. Remplacez *region* par l' Région AWS endroit où vous avez lancé votre instance temporaire, par exemple`us-east-1`.

```
wget https://aws-pcs-repo-region.s3.region.amazonaws.com/aws-pcs-slurm/aws-pcs-
slurm-25.05-installer-25.05.4-1.tar.gz.sig && \
     gpg --verify ./aws-pcs-slurm-25.05-installer-25.05.4-1.tar.gz.sig
```

La sortie doit ressembler à ce qui suit :

```
gpg: assuming signed data in './aws-pcs-slurm-25.05-installer-25.05.4-1.tar.gz'
gpg: Signature made Fri 24 Oct 2025 05:05:11 PM UTC using RSA key ID ECC0AE5C
gpg: Good signature from "AWS PCS Packages (AWS PCS Packages)"
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 1C24 32C1 862F 64D1 F90A  239A 7EEF 030E DDF5 C21C
     Subkey fingerprint: B7E1 8788 3517 6A74 C3D5  EAF5 6088 136D ECC0 AE5C
```

Si le résultat inclut `Good signature` et que l'empreinte correspond à l'empreinte renvoyée à l'étape précédente, passez à l'étape suivante.

> **Important**
>
> N'exécutez pas le script d'installation de Slurm si l'empreinte digitale ne correspond pas. Contactez AWS Support.

4. Procédez à l'extraction des fichiers à partir du fichier compressé `.tar.gz` et accédez au répertoire extrait.

```
tar -xf aws-pcs-slurm-25.05-installer-25.05.4-1.tar.gz && \
    cd aws-pcs-slurm-25.05-installer
```

5. Installez Slurm. Le programme d'installation télécharge, compile et installe Slurm et ses dépendances. Cela prend plusieurs minutes, selon les spécifications de l'instance temporaire que vous avez sélectionnée.

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

6. Consultez le fichier de version du planificateur pour confirmer l'installation.

```
cat /opt/aws/pcs/scheduler/slurm-25.05/version
```

La sortie doit ressembler à ce qui suit :

```
SLURM_INSTALL_DATE='Mon Nov 3 14:23:38 UTC 2025'
SLURM_VERSION='25.05.4'
PCS_SLURM_RELEASE='1'
```

Étape 4 — (Facultatif) Installation de pilotes, de bibliothèques et de logiciels d'application supplémentaires

Installez des pilotes, des bibliothèques et des logiciels d'application supplémentaires sur l'instance temporaire. Les procédures d'installation varient en fonction des applications et bibliothèques spécifiques. Si vous n'avez jamais créé d'AMI personnalisée pour AWS PCS auparavant, nous vous recommandons de créer et de tester d'abord une AMI en installant uniquement le logiciel AWS PCS et Slurm, puis d'ajouter progressivement vos propres logiciels et configurations une fois que vous aurez confirmé le succès initial.

Exemples

- Logiciel Elastic Fabric Adapter (EFA). Pour plus d'informations, consultez Commencer à utiliser EFA et MPI pour les charges de travail HPC sur Amazon EC2 dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.
- Client Amazon Elastic File System (Amazon EFS). Pour plus d'informations, consultez la section Installation manuelle du client Amazon EFS dans le guide de l'utilisateur Amazon Elastic File System.

- Client Lustre, pour utiliser Amazon FSx for Lustre et Amazon File Cache. Pour plus d'informations, consultez la section Installation du client Lustre dans le guide de l'utilisateur de FSx for Lustre.
- CloudWatch Agent Amazon, pour utiliser CloudWatch les journaux et les métriques. Pour plus d'informations, consultez la section Installation de l' CloudWatch agent dans le guide de CloudWatch l'utilisateur Amazon.
- AWS Neuron, pour utiliser les types d'instance trn* et inf*. Pour plus d'informations, consultez la documentation AWS Neuron.
- NVIDIA Driver, CUDA et DCGM, pour utiliser les types d'instance p* ou g*.

Étape 5 — Création d'une AMI compatible avec AWS PCS

Après avoir installé les composants logiciels requis, vous créez une AMI que vous pouvez réutiliser pour lancer des instances dans des groupes de nœuds de calcul AWS PCS.

> **Important**
>
> AWS PCS nécessite actuellement un noyau prenant en IPv4 charge la communication entre nœuds locaux, même lorsque vous utilisez AWS PCS dans un réseau IPv6 uniquement.

Pour créer une AMI à partir de votre instance temporaire

1. Ouvrez la console Amazon EC2.
2. Dans le panneau de navigation, choisissez Instances.
3. Sélectionnez l'instance temporaire que vous avez créée. Choisissez Actions, Image, Créer une image.
4. Pour Créer une image, procédez comme suit :
    a. Pour Nom de l'image, entrez un nom descriptif pour l'AMI.
    b. (Facultatif) Pour Description de l'image, saisissez une brève description de l'objectif de l'AMI.
    c. Choisissez Create image (Créer une image).
5. Dans le panneau de navigation, sélectionnez AMIs.
6. Localisez l'AMI que vous avez créée dans la liste. Attendez que son statut passe de En attente à Disponible, puis utilisez-le avec un groupe de nœuds de calcul AWS PCS.

Étape 6 — Utiliser l'AMI personnalisée avec un groupe de nœuds de calcul AWS PCS

Vous pouvez utiliser votre AMI personnalisée avec un groupe de nœuds de calcul AWS PCS nouveau ou existant.

> **Important**
>
> AWS Le PCS nécessite actuellement un noyau prenant en IPv4 charge la communication entre nœuds locaux, même lorsque vous utilisez le AWS PCS dans un réseau IPv6 uniquement.

New compute node group

Pour utiliser l'AMI personnalisée

1. Ouvrez la console AWS PCS.
2. Dans le panneau de navigation, choisissez Clusters.
3. Choisissez le cluster dans lequel vous utiliserez l'AMI personnalisée, puis sélectionnez Compute node groups.
4. Créez un nouveau groupe de nœuds de calcul. Pour de plus amples informations, veuillez consulter Création d'un groupe de nœuds de calcul dans AWS PCS. Sous ID AMI, recherchez le nom ou l'ID de l'AMI personnalisée que vous souhaitez utiliser. Terminez la configuration du groupe de nœuds de calcul, puis choisissez Créer un groupe de nœuds de calcul.
5. (Facultatif) Vérifiez que l'AMI prend en charge les lancements d'instances. Lancez une instance dans le groupe de nœuds de calcul. Vous pouvez le faire en configurant le groupe de nœuds de calcul pour qu'il n'ait qu'une seule instance statique, ou vous pouvez soumettre une tâche à une file d'attente qui utilise le groupe de nœuds de calcul.

   a. Vérifiez la console Amazon EC2 jusqu'à ce qu'une instance apparaisse étiquetée avec le nouvel ID de groupe de nœuds de calcul. Pour plus d'informations à ce sujet, voirRecherche d'instances de groupes de nœuds de calcul dans AWS PCS.

   b. Lorsque vous voyez une instance se lancer et terminer son processus d'amorçage, vérifiez qu'elle utilise l'AMI attendue. Pour ce faire, sélectionnez l'instance, puis inspectez

l'ID de l'AMI sous Détails. Elle doit correspondre à l'AMI que vous avez configurée dans les paramètres du groupe de nœuds de calcul.

c. (Facultatif) Mettez à jour la configuration de dimensionnement du groupe de nœuds de calcul selon vos valeurs préférées.

Existing compute node group

Pour utiliser l'AMI personnalisée

1. Ouvrez la console AWS PCS.
2. Dans le panneau de navigation, choisissez Clusters.
3. Choisissez le cluster dans lequel vous utiliserez l'AMI personnalisée, puis sélectionnez Compute node groups.
4. Sélectionnez le groupe de nœuds que vous souhaitez configurer et choisissez Modifier. Sous ID AMI, recherchez le nom ou l'ID de l'AMI personnalisée que vous souhaitez utiliser. Terminez la configuration du groupe de nœuds de calcul, puis choisissez Mettre à jour. Les nouvelles instances lancées dans le groupe de nœuds de calcul utiliseront l'ID AMI mis à jour. Les instances existantes continueront à utiliser l'ancienne AMI jusqu'à ce que AWS PCS les remplace. Pour de plus amples informations, veuillez consulter Mise à jour d'un groupe de nœuds de calcul AWS PCS.
5. (Facultatif) Vérifiez que l'AMI prend en charge les lancements d'instances. Lancez une instance dans le groupe de nœuds de calcul. Vous pouvez le faire en configurant le groupe de nœuds de calcul pour qu'il n'ait qu'une seule instance statique, ou vous pouvez soumettre une tâche à une file d'attente qui utilise le groupe de nœuds de calcul.

   a. Vérifiez la console Amazon EC2 jusqu'à ce qu'une instance apparaisse étiquetée avec le nouvel ID de groupe de nœuds de calcul. Pour plus d'informations à ce sujet, voirRecherche d'instances de groupes de nœuds de calcul dans AWS PCS.

   b. Lorsque vous voyez une instance se lancer et terminer son processus d'amorçage, vérifiez qu'elle utilise l'AMI attendue. Pour ce faire, sélectionnez l'instance, puis inspectez l'ID de l'AMI sous Détails. Elle doit correspondre à l'AMI que vous avez configurée dans les paramètres du groupe de nœuds de calcul.

   c. (Facultatif) Mettez à jour la configuration de dimensionnement du groupe de nœuds de calcul selon vos valeurs préférées.

Étape 7 — Mettre fin à l'instance temporaire

Après avoir confirmé que votre AMI fonctionne comme prévu avec AWS PCS, vous pouvez mettre fin à l'instance temporaire pour ne plus en payer les frais.

Pour résilier l'instance temporaire

1. Ouvrez la console Amazon EC2.
2. Dans le panneau de navigation, choisissez Instances.
3. Sélectionnez l'instance temporaire que vous avez créée et choisissez Actions, État de l'instance, Terminate instance.
4. Lorsque vous êtes invité à confirmer, choisissez Terminate.

Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS

AWS fournit un fichier téléchargeable permettant d'installer le logiciel AWS PCS sur une instance. AWS fournit également un logiciel capable de télécharger, de compiler et d'installer les versions pertinentes de Slurm et de ses dépendances. Vous pouvez utiliser ces instructions AMIs pour créer une version personnalisée à utiliser avec AWS PCS ou vous pouvez utiliser vos propres méthodes.

Table des matières

- AWS Programme d'installation du logiciel PCS Agent
- Installateur Slurm
- Systèmes d'exploitation pris en charge
- Types d'instance pris en charge
- Versions de Slurm prises en charge
- Vérifiez les installateurs à l'aide d'une somme de contrôle

AWS Programme d'installation du logiciel PCS Agent

Le programme d'installation du logiciel de l'agent AWS PCS configure une instance pour qu'elle fonctionne avec AWS PCS pendant le processus de démarrage de l'instance. Vous devez utiliser les programmes d'installation AWS fournis pour installer l'agent AWS PCS sur votre AMI personnalisée.

Pour plus d'informations sur le logiciel de l'agent AWS PCS, consultezAWS Versions de l'agent PCS.

Installateur Slurm

Le programme d'installation de Slurm télécharge, compile et installe les versions pertinentes de Slurm et de ses dépendances. Vous pouvez utiliser le programme d'installation de Slurm pour créer une version personnalisée AMIs pour AWS PCS. Vous pouvez également utiliser vos propres mécanismes s'ils sont compatibles avec la configuration logicielle fournie par le programme d'installation de Slurm. Pour plus d'informations sur le support AWS PCS pour Slurm, consultez. Versions Slurm en PCS AWS

Le logiciel AWS fourni installe les éléments suivants :

- Slurm à la version majeure et à la version de maintenance demandées (actuellement version 25.05.x) - Licence GPL 2
  - Slurm est construit avec `--sysconfdir` un set pour `/etc/slurm`
  - Slurm est conçu avec l'option et `--enable-pam` `--without-munge`
  - Slurm est conçu avec l'option `--sharedstatedir=/run/slurm/`
  - Slurm est construit avec le support PMIX et JWT
  - Slurm est installé sur `/opt/aws/pcs/schedulers/slurm-25.05`
- OpenPMIX (version 4.2.6) — Licence
  - OpenPMix est installé en tant que sous-répertoire de `/opt/aws/pcs/scheduler/`
- libjwt (version 1.17.0) — Licence MPL-2.0
  - libjwt est installé en tant que sous-répertoire de `/opt/aws/pcs/scheduler/`

Le logiciel AWS fourni modifie la configuration du système comme suit :

- Le `systemd` fichier Slurm créé par le build est copié `/etc/systemd/system/` avec le nom du fichier. `slurmd-25.05.service`
- S'ils n'existent pas, un utilisateur et un groupe Slurm (`slurm:slurm`) sont créés avec UID/GID of. `401`
- Le dossier `/etc/aws/pcs/scheduler/slurm-25.05/plugstack.conf.d/` est créé pour stocker votre Étendez les fonctionnalités de Slurm sur AWS PC avec les plugins SPANK configuration.

- Sur Amazon Linux 2 et Rocky Linux 9, l'installation ajoute le référentiel EPEL pour installer le logiciel requis pour créer Slurm ou ses dépendances.
- Lors RHEL9 de l'installation, vous pourrez activer `codeready-builder-for-rhel-9-rhui-rpms` et `epel-release-latest-9` `fedoraproject` installer le logiciel requis pour créer Slurm ou ses dépendances.

Systèmes d'exploitation pris en charge

Consultez Systèmes d'exploitation pris en charge sur AWS PCS.

> **Note**
>
> AWS Apprentissage profond (deep learning) AMIs Les versions (DLAMI) basées sur Amazon Linux 2 et Ubuntu 22.04 doivent être compatibles avec le logiciel PCS et les installateurs AWS Slurm. Pour plus d'informations, consultez la section Choix de votre DLAMI dans AWS Apprentissage profond (deep learning) AMIs le guide du développeur.

Types d'instance pris en charge

AWS Le logiciel PCS et les installateurs Slurm prennent en charge tous les types d'instances x86_64 ou arm64 capables d'exécuter l'un des systèmes d'exploitation pris en charge.

Versions de Slurm prises en charge

Consultez Versions Slurm en PCS AWS.

Vérifiez les installateurs à l'aide d'une somme de contrôle

Vous pouvez utiliser des SHA256 checksums pour vérifier les fichiers tarball du programme d'installation (.tar.gz). Nous vous recommandons de le faire pour vérifier l'identité de l'éditeur du logiciel et pour vérifier que l'application n'a pas été modifiée ou endommagée depuis sa publication.

Pour vérifier une archive tar

Utilisez l'utilitaire sha256sum pour la somme de SHA256 contrôle et spécifiez le nom du fichier tarball. Vous devez exécuter la commande depuis le répertoire dans lequel vous avez enregistré le fichier tarball.

- SHA256

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

La commande doit renvoyer une valeur de somme de contrôle au format suivant.

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

Comparez la valeur de somme de contrôle renvoyée par la commande avec la valeur de somme de contrôle fournie dans le tableau suivant. Si les sommes de contrôle correspondent, vous pouvez exécuter le script d'installation en toute sécurité.

> **Important**
>
> Si les checksums ne correspondent pas, n'exécutez pas le script d'installation. Contactez Support.

Par exemple, la commande suivante génère la SHA256 somme de contrôle pour l'archive Slurm 25.05.4-1.

```
$ sha256sum aws-pcs-slurm-25.05-installer-25.05.4-1.tar.gz
```

Exemple de sortie :

```
3b0f93bce441d4f4f6935175f2c1e81cd961cb923adb416fa6689f5592047a7d aws-pcs-slurm-25.05-installer-25.05.4-1.tar.gz
```

Les tableaux suivants répertorient les checksums des versions récentes des programmes d'installation. *us-east-1*Remplacez-le par celui Région AWS où vous utilisez le AWS PCS.

AWS Agent PCS

Installer	Télécharger le kit URL	SHA256 somme de contrôle
AWS Agent PCS 1.3.2-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/`	`06b32a952a1c849e3442e35c28ac2e4d6962`

Installer	Télécharger le kit URL	SHA256 somme de contrôle
	aws-pcs-agent/aws-pcs-agent-v1.3.2-1.tar.gz	b09286cad748f3c83d561b52ec6f
AWS Agent PCS 1.3.1-1	https://aws-pcs-repo-*us-east-1*.s3.*us-east-1*.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.3.1-1.tar.gz	5b7f1eb7b3a86bd2d331b5cb0138d868dc9452da34b480becd86af892c7e8d19
AWS Agent PCS 1.3.0-1	https://aws-pcs-repo-*us-east-1*.s3.*us-east-1*.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.3.0-1.tar.gz	eadc9b65c3db248bdde2a6c41814dfb1b97239f24ad55e03d8526dd9ab4a8d16
AWS Agent PCS 1.2.2-1	https://aws-pcs-repo-*us-east-1*.s3.*us-east-1*.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.2.2-1.tar.gz	fd7b6ea5442db75d723fc4971781ce6ae511baa21b87c4286fc1df8127b282b8
AWS Agent PCS 1.2.1-1	https://aws-pcs-repo-*us-east-1*.s3.*us-east-1*.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.2.1-1.tar.gz	2b784643ca01ccca1baa64fbfb34bb41efe8bdca69470998b74ce3962bc271d4

Installer	Télécharger le kit URL	SHA256 somme de contrôle
AWS Agent PCS 1.2.0-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.2.0-1.tar.gz`	`470db8c4fc9e50277b6317f98584b6b547e73523043e34f018eecae767846805`
AWS Agent PCS 1.1.1-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.1.1-1.tar.gz`	`bef078bf60a6d8ecde2e6c49cd34d088703f02550279e3bf483d57a235334dc6`
AWS Agent PCS 1.1.0-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.1.0-1.tar.gz`	`594c32194c71bccc5d66e5213213ae38dd2c6d2f9a950bb01accea0bbab0873a`
AWS Agent PCS 1.0.1-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.0.1-1.tar.gz`	`04e22264019837e3f42d8346daf5886eaacecd21571742eb505ea8911786bcb2`
AWS Agent PCS 1.0.0-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-agent/aws-pcs-agent-v1.0.0-1.tar.gz`	`d2d3d68d00c685435c38af471d7e2492dde5ce9eb222d7b6ef0042144b134ce0`

Installateur Slurm

Installer	Télécharger le kit URL	SHA256 somme de contrôle
Slurm 25.05.4-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-25.05-installer-25.05.4-1.tar.gz`	`3b0f93bce441d4f4f6935175f2c1e81cd961cb923adb416fa6689f5592047a7d`
Slurm 25.05.3-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-25.05-installer-25.05.3-1.tar.gz`	`851bb5815b6700ceb30cc4a3fda204ca8ce362c14528c339908983255a936cf0`
Slurm 24.11.6-2	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.11-installer-24.11.6-2.tar.gz`	`f17cd78e0bc6b9c818b794d9d2685cceabdc73f4fbb12f7566ae5b86a5abc32b`
Slurm 24.11.6-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.11-installer-24.11.6-1.tar.gz`	`225de9fc18206f5f65f412effe1fd457614ac97ee9822b3ff804a452b0fae522`
Slurm 24.11.5-1	`https://aws-pcs-repo-us-east-1.s3.us-east-1.amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.11-installer-24.11.5-1.tar.gz`	`593efe4d66bef2f3e46d5a382fb5a32f7a3ca2510bcf1b3c85739f4f951810d5`

Installer	Télécharger le kit URL	SHA256 somme de contrôle
Slurm 24.05.8-2	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.05-installer-24.05.8-2.tar.gz`	`c494b0b55c319a4c2f3faf668c759d46c32c4c7aa94ae97d94128328fe95364b`
Slurm 24.05.8-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.05-installer-24.05.8-1.tar.gz`	`210a43b376af082bbad640b2032655885790c5dab0e6489cc327c7310a375849`
Slurm 24.05.7-1	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.05-installer-24.05.7-1.tar.gz`	`0b5ed7c81195de2628c78f37c79e63fc4ae99132ca6b019b53a0d68792ee82c5`
Slurm 24.05.5-2	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-24.05-installer-24.05.5-2.tar.gz`	`7cc8d8294f2fbff95fe0602cf9e21e02003b5d96c0730e0a18c6aa04c7a4967b`
Slurm 23.11.10-4 (obsolète)	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.10-4.tar.gz`	`bb2d8c919c69dba38d14358f49c7f0427564c5dd4af85a1c9eca2c57ceeae29a`

Installer	Télécharger le kit URL	SHA256 somme de contrôle
Slurm 23.11.10-3 (obsolète)	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.10-3.tar.gz`	`488a10ee0fbd57ec0e0ff7ea708a9e3038fafdc025c6bb391c75c2e2a7852a00`
Slurm 23.11.10-2 (obsolète)	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.10-2.tar.gz`	`0bbe85423305c05987931168caf98da08a34c25f9eec0690e8e74de0b7bc8752`
Slurm 23.11.10-1 (obsolète)	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.10-1.tar.gz`	`27e8faa9980e92cdfd8cfdc71f937777f0934552ce61e33dac4ecf5a20321e44`
Slurm 23.11.9-1 (obsolète)	`https://aws-pcs-repo- us-east-1 .s3.us-east-1 .amazonaws.com/aws-pcs-slurm/aws-pcs-slurm-23.11-installer-23.11.9-1.tar.gz`	`1de7d919c8632fe8e2806611bed4fde1005a4fadc795412456e935c7bba2a9b8`

Notes de mise à jour pour un exemple de AWS PCS AMIs

AMIs pour les dernières versions majeures prises en charge du planificateur, recevez des mises à jour de sécurité et des corrections de bogues critiques. Ces correctifs de sécurité progressifs ne sont pas inclus dans les notes de publication officielles.

> **Important**
>
> AMIs Les exemples relatifs aux anciennes versions du planificateur ne sont pas pris en charge et ne reçoivent pas de mises à jour.

> **Important**
>
> AMIs Les échantillons sont fournis à des fins de démonstration et ne sont pas recommandés pour les charges de travail de production.

Table des matières

- AWS Exemple PCS AMIs pour x86_64 (Amazon Linux 2)
- AWS Exemple PCS AMIs pour Arm64 (Amazon Linux 2)

AWS Exemple PCS AMIs pour x86_64 (Amazon Linux 2)

Slurm 25,05

Nom de l'AMI

- `aws-pcs-sample_ami-amzn2-x86_64-slurm-25.05`

Instances EC2 prises en charge

- Toutes les instances dotées d'un processeur x86 64 bits. Pour trouver des instances compatibles, accédez à la console Amazon EC2. Choisissez Types d'instances, puis recherchez Architectures=x86_64.

Contenu AMI

- Service AWS pris en charge : AWS PCS
- Système d'exploitation : Amazon Linux 2
- Architecture de calcul : x86_64
- Type de volume EBS : GP2
- Installateur EFA : 1.43.1
- GDRCopy: 2,5.1
- Pilote NVIDIA : 550.127.08
- NVIDIA CUDA : 12.4.1_550.54.15

Slurm 24,11

> **Note**
>
> AWS PCS prend en charge la gestion de Slurm 24.11 et versions ultérieures. Pour de plus amples informations, veuillez consulter Comptabilité Slurm dans PCS AWS.

Nom de l'AMI

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

Instances EC2 prises en charge

- Toutes les instances dotées d'un processeur x86 64 bits. Pour trouver des instances compatibles, accédez à la console Amazon EC2. Choisissez Types d'instances, puis recherchez`Architectures=x86_64`.

Contenu AMI

- AWS Service pris en charge : AWS PCS
- Système d'exploitation : Amazon Linux 2
- Architecture de calcul : x86_64
- Type de volume EBS : GP2

- Installateur EFA : 1.33.0
- GDRCopy: 2,4
- Pilote NVIDIA : 550.127.08
- NVIDIA CUDA : 12.4.1_550.54.15

Slurm 24,05

Nom de l'AMI

- `aws-pcs-sample_ami-amzn2-x86_64-slurm-24.05`

Instances EC2 prises en charge

- Toutes les instances dotées d'un processeur x86 64 bits. Pour trouver des instances compatibles, accédez à la console Amazon EC2. Choisissez Types d'instances, puis recherchez`Architectures=x86_64`.

Contenu AMI

- AWS Service pris en charge : AWS PCS
- Système d'exploitation : Amazon Linux 2
- Architecture de calcul : x86_64
- Type de volume EBS : GP2
- Installateur EFA : 1.33.0
- GDRCopy: 2,4
- Pilote NVIDIA : 550.127.08
- NVIDIA CUDA : 12.4.1_550.54.15

Slurm 23,11

Nom de l'AMI

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

Instances EC2 prises en charge

- Toutes les instances dotées d'un processeur x86 64 bits. Pour trouver des instances compatibles, accédez à la console Amazon EC2. Choisissez Types d'instances, puis recherchez`Architectures=x86_64`.

Contenu AMI

- AWS Service pris en charge : AWS PCS
- Système d'exploitation : Amazon Linux 2
- Architecture de calcul : x86_64
- Type de volume EBS : GP2
- Installateur EFA : 1.33.0
- GDRCopy: 2,4
- Pilote NVIDIA : 550.127.08
- NVIDIA CUDA : 12.4.1_550.54.15

AWS Exemple PCS AMIs pour Arm64 (Amazon Linux 2)

Slurm 25,05

Nom de l'AMI

- `aws-pcs-sample_ami-amzn2-arm64-slurm-25.05`

Instances EC2 prises en charge

- Toutes les instances dotées d'un processeur Arm 64 bits. Pour trouver des instances compatibles, accédez à la console Amazon EC2. Choisissez Types d'instances, puis recherchez Architectures=ARM64.

Contenu AMI

- Service AWS pris en charge : AWS PCS
- Système d'exploitation : Amazon Linux 2

- Architecture informatique : arm64
- Type de volume EBS : GP2
- Installateur EFA : 1.43.1
- GDRCopy: 2,5.1
- Pilote NVIDIA : 550.127.08
- NVIDIA CUDA : 12.4.1_550.54.15

Slurm 24,11

> **Note**
>
> AWS PCS prend en charge la gestion de Slurm 24.11 et versions ultérieures. Pour de plus amples informations, veuillez consulter Comptabilité Slurm dans PCS AWS.

Nom de l'AMI

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

Instances EC2 prises en charge

- Toutes les instances dotées d'un processeur Arm 64 bits. Pour trouver des instances compatibles, accédez à la console Amazon EC2. Choisissez Types d'instances, puis recherchez`Architectures=arm64`.

Contenu AMI

- AWS Service pris en charge : AWS PCS
- Système d'exploitation : Amazon Linux 2
- Architecture informatique : arm64
- Type de volume EBS : GP2
- Installateur EFA : 1.33.0
- GDRCopy: 2,4
- Pilote NVIDIA : 550.127.08

- NVIDIA CUDA : 12.4.1_550.54.15

Slurm 24,05

Nom de l'AMI

- `aws-pcs-sample_ami-amzn2-arm64-slurm-24.05`

Instances EC2 prises en charge

- Toutes les instances dotées d'un processeur Arm 64 bits. Pour trouver des instances compatibles, accédez à la console Amazon EC2. Choisissez Types d'instances, puis recherchez`Architectures=arm64`.

Contenu AMI

- AWS Service pris en charge : AWS PCS
- Système d'exploitation : Amazon Linux 2
- Architecture informatique : arm64
- Type de volume EBS : GP2
- Installateur EFA : 1.33.0
- GDRCopy: 2,4
- Pilote NVIDIA : 550.127.08
- NVIDIA CUDA : 12.4.1_550.54.15

Slurm 23,11

Nom de l'AMI

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

Instances EC2 prises en charge

- Toutes les instances dotées d'un processeur Arm 64 bits. Pour trouver des instances compatibles, accédez à la console Amazon EC2. Choisissez Types d'instances, puis recherchez`Architectures=arm64`.

Contenu AMI

- AWS Service pris en charge : AWS PCS
- Système d'exploitation : Amazon Linux 2
- Architecture informatique : arm64
- Type de volume EBS : GP2
- Installateur EFA : 1.33.0
- GDRCopy: 2,4
- Pilote NVIDIA : 550.127.08
- NVIDIA CUDA : 12.4.1_550.54.15

Systèmes d'exploitation pris en charge sur AWS PCS

AWS PCS utilise l'Amazon Machine Image (AMI) configurée pour un groupe de nœuds de calcul afin de lancer EC2 des instances dans ce groupe de nœuds de calcul. L'AMI détermine le système d'exploitation utilisé par les EC2 instances. Vous ne pouvez pas modifier le système d'exploitation dans l'exemple AWS PCS AMIs. Vous devez créer une AMI personnalisée si vous souhaitez utiliser un autre système d'exploitation. Pour de plus amples informations, veuillez consulter Amazon Machine Images (AMIs) pour AWS PC.

Systèmes d'exploitation pris en charge

- Amazon Linux 2

  Il s'agit du système d'exploitation de l'exemple AWS PCS AMIs.

  > **Important**
  >
  > AMIs Les échantillons sont fournis à des fins de démonstration et ne sont pas recommandés pour les charges de travail de production. Vous devez créer et utiliser une AMI personnalisée pour les charges de travail de production, même si vous avez l'intention d'utiliser Amazon Linux 2.

- Amazon Linux 2023
- RedHat Linux d'entreprise 9 (RHEL 9)

  Le coût à la demande pour RHEL, quel que soit le type d'instance, est plus élevé que pour les autres systèmes d'exploitation pris en charge. Pour plus d'informations sur les tarifs, consultez les sections Tarification à la demande et Quels sont les tarifs et offres de Red Hat Enterprise Linux sur Amazon Elastic Compute Cloud ? .

- RedHat Linux d'entreprise 8 (RHEL 8)
- Rocky Linux 9

  Vous pouvez utiliser le Rocky Linux 9 officiel AMIs comme base pour une AMI personnalisée. La création de votre AMI personnalisée peut échouer si l'AMI de base ne possède pas le dernier noyau.

Pour mettre à jour le noyau

1. Lancez une instance à l'aide d'un identifiant d'AMI rocky9 à partir d'ici : https://rockylinux.org/cloud-images/
2. connectez-vous à l'instance par ssh et exécutez la commande suivante :

```
sudo yum -y update
```

3. Créez une image à partir de l'instance. Vous spécifiez cette image `ParentImage` pour votre AMI personnalisée.

- Rocky Linux 8
- Ubuntu 22.04

  Ubuntu 22.04 nécessite des clés plus sécurisées pour SSH et ne prend pas en charge les clés RSA par défaut. Nous vous recommandons de générer et d'utiliser une ED25519 clé à la place.
- Ubuntu 24.04

AWS Versions de l'agent PCS

Le logiciel de l'agent AWS PCS configure les instances EC2 lancées par AWS PCS pour une utilisation avec Slurm. Vous incluez l'agent dans une Amazon Machine Images (AMI) que vous spécifiez lorsque vous créez des groupes de nœuds de calcul pour votre cluster. Les instances EC2 lancées dans ces groupes de nœuds de calcul utilisent l'AMI spécifiée et le logiciel d'agent AWS PCS inclus. L'agent AWS PCS permet à une instance EC2 de s'enregistrer en tant que membre du cluster. Pour utiliser le dernier logiciel d'agent AWS PCS, vous devez mettre à jour votre version personnalisée AMIs. Pour plus d'informations, consultez Étape 2 — Installation de l'agent AWS PCS dans Images Amazon Machine personnalisées (AMIs) pour AWS PC.

AWS Version de l'agent PCS	Date de publication	Notes de mise à jour
v1.3.2-1	10 mars 2026	• Correction d'un problème en raison duquel les nœuds de calcul exécutant RHEL 8.10 ou Rocky Linux 8.10 ne pouvaient pas démarrer en raison d'un rétroportage `curl` SigV4 défectueux dans ces systèmes d'exploitation.
v1.3.1-1	7 novembre 2025	• Amélioration de la désactivation de l'hyperthreading en utilisant le paramètre sysfs `smt/control` lorsqu'il est disponible. • Correction d'un problème de course potentiel lorsque le processeur est verrouillé pendant le démarrage alors que l'agent PCS tente de désactiver l'hyperthreading. • Correction d'un problème en raison duquel `InstanceT`

AWS Version de l'agent PCS	Date de publication	Notes de mise à jour
		ype les champs `InstanceId` et des nœuds de calcul Slurm étaient respectivement remplis d'un horodatage et d'un trait d'union.
v1.3.0-1	3 novembre 2025	• Ajout du support pour les nouveaux systèmes d'exploitation : Amazon Linux 2023, Ubuntu 24, RHEL 8, Rocky 8.
v1.2.2-1	16 octobre 2025	• Interrogations de métadonnées d'instanc e autorisées vers un IPv6 point de terminais on si aucun IPv4 point de terminaison n'est disponible. • Correction d'un problème qui empêchait la désactiva tion de l'hyperthreading si le noyau renvoyait des threads frères sous forme de plages d'identifiants de processeur. • Correction d'un problème qui produisait de faux messages d'erreur dans les journaux lorsque l'hyperth reading était correctement désactivé.

AWS Version de l'agent PCS	Date de publication	Notes de mise à jour
v1.2.1-1	19 juin 2025	• L'agent AWS PCS essaie désormais de démarrer slurmd pendant 30 minutes maximum si la manette n'est pas disponible. • Correction d'un problème qui produisait une configuration incorrecte de slurmd si la réponse RegisterComputeNodeGroupInstance contenait un point de terminaison SLURMDBD.
v1.2.0-1	07 mars 2025	• Support activé pour IPv6 in `slurmd.conf` .
v1.1.1-1	13 décembre 2024	• Correction d'un problème en raison duquel une version incorrecte de Slurm était signalée lors de l'appel à. RegisterComputeNodeGroupInstance • Correction d'un problème en raison duquel les métadonnées de l'instance n'étaient pas extraites correctement si un script personnalisé `/opt/aws/pcs/etc/bootstrap_hooks/` était exécuté.

AWS Version de l'agent PCS	Date de publication	Notes de mise à jour
v1.1.0-1	6 décembre 2024	• Activation des scripts personnalisés `/opt/aws/pcs/etc/bootstrap_hooks/` pour qu'ils s'exécutent avant les étapes de démarrage.
v1.0.1-1	22 octobre 2024	• Correction d'un problème en raison duquel les appareils NVIDIA ne fonctionnaient pas lorsqu'ils étaient `slurmd` démarrés sur des instances dotées d'un processeur graphique.
v1.0.0-1	28 août 2024	• Première version.

Planificateur Slurm en PCS AWS

Slurm est un gestionnaire de charge de travail open source conçu pour les clusters Linux qui fournit des fonctionnalités de planification des tâches, d'allocation de ressources et de surveillance des tâches pour les charges de travail HPC. AWS PCS prend en charge le planificateur Slurm pour gérer les charges de travail de votre cluster.

Rubriques

- Versions Slurm en PCS AWS
- Comptabilité Slurm dans PCS AWS
- API REST Slurm sur PCS AWS
- Redémarrage de nœuds de calcul avec Slurm sur PCS AWS
- Configuration des paramètres personnalisés de Slurm dans PCS AWS
- Étendez les fonctionnalités de Slurm sur AWS PC avec les plugins SPANK
- Utilisez les plugins de filtre Slurm CLI pour personnaliser la soumission des tâches dans PCS AWS

Versions Slurm en PCS AWS

SchedMD améliore continuellement Slurm avec de nouvelles fonctionnalités, optimisations et correctifs de sécurité. SchedMD publie une nouvelle version majeure à intervalles réguliers et prévoit de prendre en charge jusqu'à 3 versions à la fois. AWS Le PCS est conçu pour mettre à jour automatiquement le contrôleur Slurm avec des versions de patch.

Lorsque SchedMD met fin au support d'une version majeure particulière, AWS PCS désigne cette version comme étant en fin de vie (EOL). Après EOL, aucun nouveau cluster ne peut être créé avec cette version, bien que les clusters existants puissent continuer à fonctionner jusqu'à 12 mois sans garantie de support. AWS PCS envoie un préavis si une version majeure de Slurm est proche de la fin de vie, afin d'aider les clients à savoir quand mettre à niveau leurs clusters vers une version plus récente prise en charge.

Nous vous recommandons d'utiliser la dernière version prise en charge de Slurm pour déployer votre cluster, afin d'accéder aux avancées et améliorations les plus récentes.

Versions de Slurm prises en charge sur PCS AWS

Le tableau suivant indique les versions de Slurm prises en charge ainsi que les dates et informations importantes pour chaque version.

Version Slurm	Date de sortie de SchedMD	AWS Date de sortie du PCS	AWS Date EOL du PCS	Version minimale de l'agent AWS PCS compatible	Exemple de AWS PCS pris en charge AMIs
25,05	29/05/2025	16/10/2025	31/05/2027	1.0.0-1	• `aws-pcs-sample_ami-amzn2-x86_64-slurm-25.05` • `aws-pcs-sample_ami-amzn2-arm64-slurm-25.05`
24,11	29/11/2024	14/05/2025	31/05/2026	1.0.0-1	• `aws-pcs-sample_ami-amzn2-x86_64-slurm-24.11` • `aws-pcs-sample_ami`

Version Slurm	Date de sortie de SchedMD	AWS Date de sortie du PCS	AWS Date EOL du PCS	Version minimale de l'agent AWS PCS compatible	Exemple de AWS PCS pris en charge AMIs
					`-amzn2-arm64-slurm-24.11`

Versions de Slurm non prises en charge sur PCS AWS

Le tableau suivant indique les versions de Slurm qui ne sont pas prises en AWS charge par PCS.

Version Slurm	Date de sortie de SchedMD	AWS Date de sortie du PCS	AWS Date EOL du PCS		
24,05	30/05/2024	18/12/2024	30/11/2025		
23,11	21/11/2023	28/08/2024	31/05/2025		

Notes de mise à jour pour les versions de Slurm sur PCS AWS

Cette rubrique décrit les modifications importantes apportées à chaque version de Slurm actuellement prise en AWS charge par PCS. Nous vous recommandons de vérifier les modifications entre l'ancienne et la nouvelle version lors de la mise à niveau de votre cluster.

Slurm 25,05

Changements mis en œuvre dans AWS PCS

- Le Slurm requeue_on_resume_failure est désormais activé par défaut SchedulerParameter .
- « stderr » a été supprimé en tant qu'option pour LogTimeFormat, car il était désactivé dans Slurm 25.05.

- AWS PCS prend en charge la configuration sackd multi-clusters : le nœud de connexion peut accéder à plusieurs clusters.

Pour plus d'informations sur Slurm 25.05, consultez les publications suivantes :

- Annonce de sortie de SchedMD : https://www.schedmd.com/slurm-version-25-05-0-is-now-available/
- Notes de mise à jour de SchedMD : _Notes.md https://github.com/SchedMD/ slurm/blob/ slurm-25-05-0-1/RELEASE

Slurm 24,11

Changements mis en œuvre dans AWS PCS

- AWS PCS prend en charge la comptabilité Slurm. Pour de plus amples informations, veuillez consulter Comptabilité Slurm dans PCS AWS.

Pour plus d'informations sur Slurm 24.11, consultez les publications suivantes :

- Annonce de sortie de SchedMD
- Notes de mise à jour de SchedMD

Slurm 24,05

Changements mis en œuvre dans AWS PCS

- Le nouveau module Slurm Step Manager est désormais activé par défaut dans AWS PCS. Ce module offre des avantages significatifs en déléguant la gestion des étapes du contrôleur central aux nœuds de calcul, améliorant ainsi considérablement la simultanéité du système dans les environnements où l'utilisation d'étapes est importante. Pour prendre en charge cette configuration et améliorer l'isolation `Prolog` et l'exécution des `Epilog` processus, de nouveaux indicateurs de prolog (`Contain`,`Alloc`) sont activés.
- La communication hiérarchique entre le contrôleur et les nœuds de calcul est activée pour optimiser la communication intra-nœud de Slurm, ce qui améliore l'évolutivité et les performances. En outre, la configuration de routage utilise désormais des listes de nœuds de partition pour les communications provenant du contrôleur, au lieu de l'algorithme de routage par défaut du plugin, ce qui améliore la résilience du système.

- Un nouveau plugin de hachage `HashPlugin=hash/sha3` remplace le précédent`hash/k12` `plugin`. Ceci est désormais activé par défaut dans les clusters AWS PCS.
- Les journaux du contrôleur Slurm incluent désormais des fonctionnalités d'audit améliorées pour tous les appels de procédure à distance (RPC) entrants adressés à. `slurmctld` Les journaux incluent l'adresse source, l'utilisateur authentifié et le type RPC avant le traitement de la connexion.

Pour plus d'informations sur Slurm 24.05, consultez les publications suivantes :

- Annonce de sortie de SchedMD
- Notes de mise à jour de SchedMD

Slurm 23,11

Réglages de Slurm que vous pouvez modifier dans PCS AWS

- La `SuspendTime` valeur par défaut est. `60` Utilisez le paramètre `scaleDownIdleTimeInSeconds` de configuration AWS PCS pour le définir. Pour plus d'informations, consultez le `scaleDownIdleTimeInSeconds`paramètre du type de `ClusterSlurmConfiguration` données dans la référence de l'API AWS PCS.
- Le `MaxJobCount` et `MaxArraySize` est basé sur la taille que vous avez choisie pour le cluster. Pour plus d'informations, consultez le `size`paramètre de l'action d'`CreateCluster`API dans la référence d'API AWS PCS.
- Le paramètre `SelectTypeParameters` Slurm est défini par défaut sur. `CR_CPU` Vous pouvez le fournir sous forme de valeur `slurmCustomSettings` pour le définir lorsque vous créez un cluster. Pour plus d'informations, consultez le `slurmCustomSettings`paramètre de l'action d'`CreateCluster`API et le manuel SlurmCustomSettingde référence de l'API AWS PCS.
- Vous pouvez définir `Prolog` et `Epilog` au niveau du cluster. Vous pouvez le fournir sous forme de valeur `slurmCustomSettings` pour le définir lorsque vous créez un cluster. Pour plus d'informations, voir `CreateCluster`et SlurmCustomSettingdans le manuel de référence de l'API AWS PCS.
- Vous pouvez définir `Weight` et `RealMemory` au niveau du groupe de nœuds de calcul. Vous pouvez le fournir sous forme de valeur `slurmCustomSettings` pour le définir lorsque vous créez un groupe de nœuds de calcul. Pour plus d'informations, voir `CreateComputeNodeGroup`et SlurmCustomSettingdans le manuel de référence de l'API AWS PCS.

Questions fréquemment posées sur les versions de Slurm dans PCS AWS

AWS PCS maintient le support pour plusieurs versions de Slurm. Lorsqu'une nouvelle version de Slurm est introduite, AWS PCS fournit un support technique et des correctifs de sécurité jusqu'à ce que cette version atteigne la fin du support (EOS) de SchedMD. AWS Par souci de cohérence avec la terminologie, PCS désigne la date EOS d'une version de Slurm comme étant la date de fin de vie (EOL). AWS

Pendant combien de temps AWS PCS supporte-t-il une version de Slurm ?

AWS Le support PCS pour les versions de Slurm s'aligne sur les cycles de support de SchedMD pour les versions majeures. AWS PCS prend en charge la version actuelle et les 2 versions majeures précédentes les plus récentes. Lorsque SchedMD publie une nouvelle version majeure, AWS PCS met fin au support de la version supportée la plus ancienne. AWS PCS publie de nouvelles versions majeures de Slurm dès que possible, mais il se peut qu'il y ait un délai entre la sortie de SchedMD et sa disponibilité sur PCS. AWS

Comment mes clusters bénéficient-ils des nouvelles versions de correctif de Slurm ?

Pour corriger les bogues et corriger les problèmes de sécurité, le AWS PCS est conçu pour appliquer automatiquement des correctifs aux contrôleurs de cluster qui s'exécutent sur des comptes appartenant au service interne. Pour installer des correctifs sur vos instances EC2 Compte AWS, mettez à jour l'Amazon Machine Image (AMI) pour vos groupes de nœuds de calcul et mettez à jour les groupes de nœuds de calcul afin d'utiliser l'AMI mise à jour. Pour de plus amples informations, veuillez consulter Images Amazon Machine personnalisées (AMIs) pour AWS PC.

> **Note**
>
> Les manettes Slurm ne sont pas disponibles pendant leur mise à jour. Les tâches en cours ne sont pas affectées. Les tâches soumises avant que le contrôleur du cluster ne soit indisponible sont conservées jusqu'à ce que le contrôleur soit disponible.

Comment suis-je informé d'un prochain événement EOL pour la version Slurm ?

Nous vous envoyons un e-mail 6 mois avant la date de fin de vie. Nous vous envoyons un e-mail chaque mois avant la fin de vie, avec un dernier e-mail une semaine avant la date de fin de vie. Après la date d'expiration, nous envoyons des e-mails mensuels pendant 12 mois aux clients utilisant des clusters AWS PCS avec des versions EOL Slurm. Nous pouvons suspendre un cluster doté d'une version EOL Slurm si des failles de sécurité sont identifiées pour cette version.

Comment puis-je déterminer si la version de Slurm utilisée par mon cluster exécute une version EOL Slurm ?

Nous vous envoyons un e-mail pour vous informer que vous avez un cluster en cours d'exécution avec une version EOL Slurm. Nous publions une Tableau de bord AWS Health alerte contenant les détails de vos clusters avec les versions d'EOL Slurm. Vous pouvez également utiliser la console AWS PCS pour identifier les clusters dotés de versions EOL Slurm.

Que dois-je faire si ma version de Slurm est proche ou supérieure à la fin de sa vie ?

Créez un nouveau cluster avec une nouvelle version prise en charge de Slurm et mettez à jour la version Slurm dans les AMI de votre groupe de nœuds de calcul. La version de Slurm présente dans vos AMI et les instances EC2 en cours d'exécution ne peuvent pas avoir plus de 2 versions de retard par rapport à la version Slurm du cluster. Pour de plus amples informations, veuillez consulter Images Amazon Machine personnalisées (AMIs) pour AWS PC.

Que se passera-t-il si je ne passe pas à une version plus récente de Slurm avant la date de fin de vie ?

Vous ne pouvez pas créer de nouveaux clusters avec une version EOL Slurm. Les clusters existants peuvent fonctionner jusqu'à 12 mois sans AWS assistance, et aucune action immédiate n'est requise pour maintenir leur fonctionnement. Après la date d'expiration, le support, les mises à jour de sécurité et la disponibilité ne sont pas garantis. Nous pouvons suspendre un cluster pour des raisons de sécurité. Nous vous recommandons vivement d'utiliser une version compatible de Slurm pour garantir la sécurité et le support de vos clusters AWS PCS.

Quels sont les risques liés à l'exploitation d'un cluster avec les versions EOL Slurm ?

Les clusters dotés de versions EOL Slurm présentent des risques opérationnels et de sécurité importants. Sans la surveillance active de SchedMD, les failles de sécurité risquent de ne pas être détectées ou de ne pas être corrigées. Si des vulnérabilités critiques sont découvertes, nous pouvons suspendre immédiatement vos clusters.

Qu'arrive-t-il à mes tâches, aux ressources de calcul, de stockage et de réseau de mon cluster lorsque mon cluster est suspendu ?

Toutes les ressources gérées par AWS PCS sont supprimées. Cela inclut le contrôleur Slurm, les groupes de nœuds de calcul et les instances EC2. Toutes les tâches exécutées sur des instances de calcul sont immédiatement interrompues et le cluster entre dans un état suspendu. Les ressources gérées par le client, telles que les systèmes de fichiers externes, restent intactes. Vous pouvez utiliser la console AWS PCS et les actions de l'API pour accéder à la configuration du cluster.

Puis-je redémarrer un cluster suspendu pour reprendre ses tâches restantes ?

Non, vous ne pouvez pas redémarrer un cluster suspendu. Vous pouvez utiliser la configuration de votre cluster suspendu pour créer un nouveau cluster avec une version compatible de Slurm. Vous pouvez exécuter les tâches restantes si vous les avez enregistrées dans un système de fichiers externe.

Puis-je demander une prolongation au-delà de la période de grâce de 12 mois ?

Non, vous ne pouvez pas demander de prolongation pour exécuter votre cluster au-delà de la période de grâce de 12 mois. Nous vous fournissons le délai supplémentaire pour vous aider à passer à une version compatible de Slurm. Pour éviter d'interrompre les opérations de votre cluster, nous vous recommandons de changer avant que votre version de Slurm n'atteigne la fin de vie.

Comptabilité Slurm dans PCS AWS

Vous pouvez activer la comptabilité sur vos nouveaux clusters AWS PCS pour surveiller l'utilisation des clusters, appliquer des limites de ressources et gérer un contrôle d'accès précis à des files d'attente ou à des groupes de nœuds de calcul spécifiques. AWS PCS crée et gère la base de données comptable de votre cluster, vous évitant ainsi de devoir créer et gérer votre propre base de données comptable distincte. AWS PCS utilise la fonction de comptabilité de Slurm. Pour plus d'informations sur la fonctionnalité de comptabilité dans Slurm, consultez la documentation de Slurm sur SchedMD.

Pour utiliser la comptabilité, activez-la lorsque vous créez un nouveau cluster et définissez éventuellement les paramètres de comptabilité. Une fois que le statut de votre cluster est `Active` défini et qu'il comporte des groupes de nœuds de calcul, vous pouvez vous connecter au shell Linux d'un nœud de connexion pour exécuter des fonctions de comptabilité, telles que l'affichage des données des tâches à l'aide de la commande Slurm`sacct`.

> **Note**
>
> La comptabilité est prise en charge pour Slurm 24.11 ou version ultérieure.

AWS PCS console

Sur la page Créer un cluster, vous devez sélectionner une version valide de Slurm (version 24.11 ou ultérieure). Dans les paramètres du planificateur, activez la comptabilité.

AWS PCS API

Fournissez la `accounting` configuration dans votre appel à l'action d'`CreateCluster`API. Dans l'`accounting`objet, définissez la valeur `mode` sur`STANDARD`. Pour plus d'informations, consultez la section CreateCluster« Comptabilité » dans le manuel de référence de l'API AWS PCS.

L'exemple suivant utilise le AWS CLI pour appeler l'action d'`CreateCluster`API. La sous-chaîne de valeur du paramètre `accounting='{mode=STANDARD}'` active la comptabilité.

```
aws pcs create-cluster --cluster-name cluster-name \
                       --scheduler type=SLURM,version=24.11 \
                       --size SMALL \
                       --networking subnetIds=cluster-subnet-
id,securityGroupIds=cluster-security-group-id \
                       --slurm-configuration
 scaleDownIdleTimeInSeconds=180,accounting='{mode=STANDARD}',slurmCustomSettings='[{paramete
```

> **Important**
>
> Des frais de facturation supplémentaires vous seront facturés si vous activez la comptabilité. Pour plus d'informations, consultez la page de tarification du AWS PCS.

Modification des paramètres comptables

Vous pouvez activer ou désactiver la comptabilité sur les clusters existants sans avoir à reconstruire votre infrastructure. Pour de plus amples informations, veuillez consulter Mettre à jour un cluster dans AWS PCS.

Lorsque vous désactivez la comptabilité, la facturation de la fonctionnalité de comptabilité s'arrête dès que le cluster entre dans l'`UPDATING`état. Lorsque vous activez la comptabilité, la facturation commence lorsque le cluster revient avec succès à l'`ACTIVE`état.

Concepts clés pour la comptabilité Slurm dans PCS AWS

Les concepts suivants sont spécifiques au AWS PCS et contrôlent la manière dont le AWS PCS implémente la comptabilité Slurm.

Base de données de comptabilité

AWS PCS stocke vos données comptables dans une base de données créée dans un Compte AWS AWS propriétaire. Vous n'avez pas accès au `slurmdbd.conf`.

Heure de purge par défaut

Ce paramètre AWS PCS spécifie la période de conservation (en jours) pour tous les types d'enregistrements comptables (tâches, événements, réservations, étapes, suspensions, transactions, données d'utilisation). Par exemple, si la valeur est 30, AWS PCS conserve les enregistrements comptables pendant 30 jours. Vous fournissez cette valeur lorsque vous créez le cluster. Si vous ne fournissez aucune valeur, AWS PCS conserve les enregistrements comptables dans la base de données indéfiniment.

AWS PCS console

Vous spécifiez l'heure de purge par défaut dans le cadre des étapes de création d'un cluster. Sur la page Créer un cluster, vous devez sélectionner une version valide de Slurm (version 24.11 ou ultérieure) et activer la gestion des comptes. Dans les paramètres du planificateur, fournissez une valeur entière pour le délai de purge par défaut (jours).

AWS PCS API

Spécifiez-le dans le `defaultPurgeTimeInDays` cadre des `accounting` informations que vous fournissez dans votre appel à l'action d'`CreateCluster`API. Pour plus d'informations, consultez la section CreateCluster« Comptabilité » dans le manuel de référence de l'API AWS PCS.

> **Note**
>
> Lorsque vous utilisez l'API AWS PCS pour créer un cluster, la valeur par défaut `defaultPurgeTimeInDays` est valide `-1` et `0` n'est pas une valeur valide.

Application des politiques comptables

Ce paramètre détermine la rigueur avec laquelle Slurm applique les règles de soumission des tâches, les limites de ressources et les politiques comptables pour votre cluster. Ce paramètre correspond au `AccountingStorageEnforce` paramètre du `slurm.conf` fichier de votre cluster. Vous pouvez sélectionner n'importe quelle combinaison d'options d'application. Si vous ne sélectionnez aucune

option, aucune contrainte comptable n'est appliquée aux tâches du cluster. AWS PCS prend en charge les options suivantes :

- associations — job-to-account cartographie
- limites — contraintes en matière de ressources
- QoS : exigences en matière de qualité de service
- mode sécurisé : achèvement garanti dans les limites
- nosteps — désactive la comptabilisation des étapes
- nojobs — désactive la comptabilité des tâches

Pour plus d'informations sur ces options, consultez la documentation de Slurm sur SchedMD.

AWS PCS console

Vous définissez les options dans le cadre des étapes de création d'un cluster. Sur la page Créer un cluster, vous devez sélectionner une version valide de Slurm (version 24.11 ou ultérieure) et activer la gestion des comptes. Sélectionnez les options souhaitées dans la liste déroulante Application des règles comptables sous Paramètres du planificateur.

AWS PCS API

Dans Slurm, ces options sont définies dans le fichier d'un cluster. `slurm.conf` Vous n'avez pas d'accès direct au cluster `slurm.conf` pour votre AWS PCS. Au lieu de cela, vous fournissez une action `SlurmCustomSettings` à l'`CreateCluster`API lorsque vous créez un cluster. Pour plus d'informations, consultez CreateClusterla référence de l'API AWS PCS.

Obtenez la configuration comptable d'un cluster AWS PCS existant

La configuration comptable Slurm est incluse dans la configuration Slurm de votre cluster.

AWS PCS console

1. Choisissez Clusters dans le volet de navigation.
2. Choisissez le nom du cluster dans la liste.
3. Dans l'onglet Configuration, trouvez la configuration comptable sous Configuration de Slurm

AWS PCS API

Utilisez l'action `GetCluster` API pour obtenir la configuration du cluster. Vous trouverez la configuration comptable dans le`slurmConfiguration`. Le paramètre `mode` et la valeur de `defaultPurgeTimeInDays` sont inférieurs à`accounting`. Les options d'application des politiques comptables sélectionnées se trouvent ci-dessous`slurmCustomSettings`. Pour plus d'informations, consultez GetClusterla référence de l'API AWS PCS.

API REST Slurm sur PCS AWS

AWS PCS fournit un support géré pour l'API REST native de Slurm via `slurmrestd` une interface HTTP pour l'interaction programmatique avec les clusters. Vous pouvez soumettre des tâches, surveiller l'état du cluster et gérer les ressources par le biais de requêtes HTTP standard sans avoir besoin d'un accès shell direct à votre cluster.

Cas d'utilisation courants

L'API REST de Slurm prend en charge différents scénarios d'intégration :

- Intégration d'applications Web : créez des interfaces personnalisées et des applications Web qui soumettent et gèrent directement les tâches.
- Intégration à Jupyter Notebook : permet aux chercheurs de soumettre des tâches à partir d'environnements de blocs-notes sans quitter leur flux de travail de développement.
- Intégration de solutions partenaires : connectez des outils HPC et des gestionnaires de flux de travail tiers à vos clusters AWS PCS.
- Gestion programmatique des clusters : automatisez les flux de travail de soumission des tâches, de surveillance et de gestion des ressources.
- Flux de travail informatiques pour la recherche : Supportez les environnements de recherche universitaires et d'entreprise qui nécessitent une gestion des tâches basée sur des API.

Exigences et limitations

Avant d'utiliser l'API REST de Slurm, consultez les informations suivantes :

- Votre cluster doit utiliser la version 25.05 ou supérieure de Slurm.
- Le point de terminaison de l'API ne sera accessible que via une adresse IP privée au sein du VPC de votre cluster.

- Le groupe de sécurité de votre cluster doit autoriser le trafic HTTP sur le port 6820.
- L'authentification nécessite des jetons JWT avec des revendications d'identité utilisateur spécifiques.

Les limites actuelles incluent :

- Les jetons générés par ne `scontrol token` sont pas pris en charge.
- `X-SLURM-USER-NAME`l'usurpation d'identité d'en-tête n'est pas disponible.
- Certaines fonctionnalités nécessitent l'activation de la comptabilité Slurm.
- Non compatible avec le mécanisme du plugin de filtre Slurm CLI.
- Les connexions au point de terminaison de l'API REST ne sont pas chiffrées avec TLS.

Rubriques

- Activation de l'API REST Slurm sur PCS AWS
- Authentification avec l'API REST de Slurm sur PCS AWS
- Utilisation de l'API REST de Slurm pour la gestion des tâches sur PCS AWS
- Questions fréquemment posées sur l'API REST de Slurm sur PCS AWS

Activation de l'API REST Slurm sur PCS AWS

Activez l'API REST de Slurm pour accéder à l'interface HTTP de votre cluster pour la gestion et la surveillance programmatiques des tâches. Vous pouvez activer cette fonctionnalité lors de la création du cluster ou mettre à jour un cluster existant qui répond aux exigences.

Conditions préalables

Avant d'activer l'API REST de Slurm, assurez-vous d'avoir :

- Version du cluster : Slurm version 25.05 ou supérieure.
- Groupe de sécurité : règles autorisant le trafic HTTP sur le port 6820 à partir des sources souhaitées.

Procédure

Pour activer l'API REST de Slurm sur un nouveau cluster

AWS Management Console

1. Ouvrez la console AWS PCS à l'adresse https://console.aws.amazon.com/pcs/.
2. Choisissez Créer un cluster.
3. Sous Détails du cluster, choisissez Slurm version 25.05 ou supérieure.
4. Configurez les autres paramètres du cluster selon vos besoins.
5. Dans la section Configuration du planificateur, définissez l'API REST sur Activé.
6. Configurez le groupe de sécurité de votre cluster pour autoriser le trafic HTTP sur le port 6820 à partir des sources souhaitées.
7. Terminez le processus de création du cluster.

AWS CLI

1. Ajoutez une configuration Slurm REST lors de la création de votre cluster.

```
aws pcs create-cluster --region region \
    --cluster-name my-cluster \
    --scheduler type=SLURM, version=25.05 \
    --size SMALL \
    --networking subnetIds=subnet-ExampleId1,securityGroupIds=sg-ExampleId1 \
    --slurm-configuration slurmRest='{mode=STANDARD}'
```

2. Configurez le groupe de sécurité de votre cluster pour autoriser le trafic HTTP sur le port 6820 à partir des sources souhaitées.

Pour activer l'API REST de Slurm sur un cluster existant

AWS Management Console

1. Ouvrez la console AWS PCS à l'adresse https://console.aws.amazon.com/pcs/.
2. Choisissez votre cluster dans la liste.
3. Vérifiez que votre cluster utilise la version 25.05 ou supérieure de Slurm dans les détails du cluster.
4. Choisissez Modifier le cluster.
5. Dans la section Configuration du planificateur, définissez l'API REST sur Activé.
6. Choisissez Mettre à jour le cluster pour appliquer les modifications.

7. Configurez le groupe de sécurité de votre cluster pour autoriser le trafic HTTP sur le port 6820 à partir des sources souhaitées.

AWS CLI

1. Mettez à jour votre cluster avec une configuration Slurm REST, comme dans cet exemple.

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

2. Configurez le groupe de sécurité de votre cluster pour autoriser le trafic HTTP sur le port 6820 à partir des sources souhaitées.

Que se passe-t-il après l'activation

Lorsque vous activez l'API REST, AWS PCS effectue automatiquement les opérations suivantes :

- Génère une clé de signature JWT et la stocke dans AWS Secrets Manager.
- Expose le point de terminaison de l'API au `https://<clusterPrivateIpAddress>:6820` sein de votre VPC.
- Met à jour la configuration de votre cluster pour afficher les détails du point de terminaison de l'API REST.

Vous pouvez désormais vous authentifier et utiliser l'API REST pour la gestion des tâches et les opérations de cluster.

Authentification avec l'API REST de Slurm sur PCS AWS

L'API REST de Slurm dans AWS PCS utilise l'authentification JSON Web Token (JWT) pour garantir un accès sécurisé aux ressources de votre cluster. AWS PCS fournit une clé de signature gérée stockée dans AWS Secrets Manager, que vous utilisez pour générer des jetons JWT contenant des revendications d'identité utilisateur spécifiques.

Conditions préalables

Avant de vous authentifier avec l'API REST de Slurm, assurez-vous d'avoir :

- Configuration du cluster : cluster AWS PCS avec Slurm 25.05+ et API REST activée.

- Autorisations AWS : accès à AWS Secrets Manager pour la clé de signature JWT.
- Informations utilisateur : nom d'utilisateur, ID utilisateur POSIX et un ou plusieurs groupes POSIX IDs pour votre compte de cluster.
- Accès réseau : connectivité au sein du VPC de votre cluster avec le groupe de sécurité autorisant le port 6820.

Procédure

Pour récupérer l'adresse du point de terminaison de l'API REST Slurm

AWS Management Console

1. Ouvrez la console AWS PCS à l'adresse https://console.aws.amazon.com/pcs/.
2. Choisissez votre cluster dans la liste.
3. Dans les détails de configuration du cluster, recherchez la section Endpoints.
4. Notez l'adresse IP privée et le port de l'API REST de Slurm (slurmrestd).
5. Vous pouvez effectuer des appels d'API en envoyant des requêtes HTTP correctement formatées à cette adresse.

AWS CLI

1. Demandez l'état de votre cluster avec`aws pcs get-cluster`. Recherchez le `SLURMRESTD` point final dans le `endpoints` champ de la réponse. Voici un exemple :

```
"endpoints": [
      {
          "type": "SLURMCTLD",
          "privateIpAddress": "192.0.2.1",
          "port": "6817"
      },
      {
          "type": "SLURMRESTD",
          "privateIpAddress": "192.0.2.1",
          "port": "6820"
      }
  ]
```

2. Vous pouvez effectuer des appels d'API en envoyant des requêtes HTTP correctement formatées à `http://<privateIpAddress>:<port>/`

Pour récupérer la clé de signature JWT

1. Ouvrez la console AWS PCS à l'adresse https://console.aws.amazon.com/pcs/.
2. Choisissez votre cluster dans la liste.
3. Dans les détails de configuration du cluster, recherchez la section Authentification du planificateur.
4. Notez l'ARN et la version de la clé JSON Web Token (JWT).
5. Utilisez le AWS CLI pour récupérer la clé de signature depuis Secrets Manager :

```
aws secretsmanager get-secret-value --secret-
id arn:aws:secretsmanager:region:account:secret:name --version-id version
```

Pour générer un jeton JWT

1. Créez un JWT avec les revendications requises suivantes :
   - `exp`— Délai d'expiration en secondes depuis 1970 pour le JWT
   - `iat`— Heure actuelle en secondes depuis 1970
   - `sun`— Le nom d'utilisateur pour l'authentification
   - `uid`— L'ID utilisateur POSIX
   - `gid`— L'identifiant du groupe POSIX
   - `id`— Propriétés d'identité POSIX supplémentaires
     - `gecos`— Champ de commentaire utilisateur, souvent utilisé pour stocker un nom lisible par l'homme
     - `dir`— Répertoire personnel de l'utilisateur
     - `shell`— Shell par défaut de l'utilisateur
     - `gids`— Liste des groupes POSIX supplémentaires auxquels IDs appartient l'utilisateur
2. Signez le JWT à l'aide de la clé de signature récupérée dans Secrets Manager.
3. Définissez une date d'expiration appropriée pour le jeton.

> **Note**
>
> Comme alternative à la `sun` réclamation, vous pouvez fournir l'un des documents suivants :
>
> - `username`
> - Un nom de champ personnalisé que vous définissez via `userclaimfield` le `AuthAltParameters Slurm custom settings`
> - Un `name` champ dans la `id` réclamation

Pour authentifier les demandes d'API

1. Incluez le jeton JWT dans vos requêtes HTTP en utilisant l'une des méthodes suivantes :
   - jeton porteur — Ajouter un en-tête `Authorization: Bearer` *<jwt>*
   - En-tête Slurm — Ajouter un en-tête `X-SLURM-USER-TOKEN:` *<jwt>*
2. Envoyez des requêtes HTTP au point de terminaison de l'API REST :

   Voici un exemple d'accès à l'`/ping`API à l'aide de curl et de l'`Authorized: Beareren`-tête.

   ```
   curl -X GET -H "Authorization: Bearer <jwt>" \
         http://<privateIpAddress>:6820/slurm/v0.0.43/ping
   ```

Exemple de génération JWT

Récupérez la clé de signature JWT du cluster AWS PCS et stockez-la dans un fichier local. Remplacez les valeurs pour aws-region, secret-arn et secret version par des valeurs adaptées à votre cluster.

```
#!/bin/bash
SECRET_KEY=$(aws secretsmanager get-secret-value \
  --region aws-region \
  --secret-id secret-arn \
  --version-stage secret-version \
  --query 'SecretString' \
  --output text)
echo "$SECRET_KEY" | base64 --decode > jwt.key
```

Cet exemple Python illustre comment utiliser la clé de signature pour générer un jeton JWT :

```
#!/usr/bin/env python3

import sys
import os
import pprint
import json
import time
from datetime import datetime, timedelta, timezone
from jwt import JWT
from jwt.jwa import HS256
from jwt.jwk import jwk_from_dict
from jwt.utils import b64decode,b64encode
if len(sys.argv) != 3:
    sys.exit("Usage: gen_jwt.py [jwt_key_file] [expiration_time_seconds]")
SIGNING_KEY = sys.argv[1]
EXPIRATION_TIME = int(sys.argv[2])
with open(SIGNING_KEY, "rb") as f:
    priv_key = f.read()
signing_key = jwk_from_dict({
    'kty': 'oct',
    'k': b64encode(priv_key)
})
message = {
    "exp": int(time.time() + EXPIRATION_TIME),
    "iat": int(time.time()),
    "sun": "ec2-user",
    "uid": 1000,
    "gid": 1000,
    "id": {
        "gecos": "EC2 User",
        "dir": "/home/ec2-user",
        "gids": [1000],
        "shell": "/bin/bash"
    }
}
a = JWT()
compact_jws = a.encode(message, signing_key, alg='HS256')
print(compact_jws)
```

Le script imprimera un JWT à l'écran.

```
abcdefgtjwttoken...
```

Utilisation de l'API REST de Slurm pour la gestion des tâches sur PCS AWS

Présentation de l'API REST de Slurm

L'API REST de Slurm fournit un accès programmatique aux fonctions de gestion des clusters par le biais de requêtes HTTP. La compréhension de ces caractéristiques clés vous aidera à utiliser efficacement l'API avec AWS PCS :

- Protocole d'accès : L'API utilise le protocole HTTP (et non HTTPS) pour les communications au sein du réseau privé de votre cluster.
- Détails de connexion : accédez à l'API à l'aide de l'adresse IP privée de votre cluster et du `slurmrestd` port (généralement 6820). Le format d'URL de base complet est`http://<privateIpAddress>:6820`.
- Versionnage de l'API : La version de l'API correspond à votre installation de Slurm. Pour Slurm 25.05, utilisez la version v0.0.43. Le numéro de version change à chaque sortie de Slurm. Vous trouverez les versions d'API actuellement prises en charge dans les notes de mise à jour de Slurm.
- Structure d'URL : La structure d'URL de l'API REST de Slurm est. `http://<privateIpAddress>:<port>/<api-version>/<endpoint>` Vous trouverez des informations détaillées sur l'utilisation des points de terminaison de l'API REST dans la documentation de Slurm.

Conditions préalables

Avant d'utiliser l'API REST de Slurm, assurez-vous d'avoir :

- Configuration du cluster : cluster AWS PCS avec Slurm 25.05+ et API REST activée.
- Authentification : jeton JWT valide avec des demandes d'identité utilisateur appropriées.
- Accès réseau : connectivité au sein du VPC de votre cluster avec un groupe de sécurité autorisant le port 6820.

Procédure

Pour soumettre une tâche à l'aide de l'API REST

1. Créez une demande de soumission de travail avec les paramètres requis :

```
{
  "job": {
    "name": "my-job",
    "partition": "compute",
    "nodes": 1,
    "tasks": 1,
    "script": "#!/bin/bash\necho 'Hello from Slurm REST API'"
  }
}
```

2. Soumettez la tâche à l'aide d'une requête HTTP POST :

```
curl -X POST \
  -H "Authorization: Bearer <jwt>" \
  -H "Content-Type: application/json" \
  -d '<job-json>' \
  https://<privateIpAddress>:6820/slurm/v0.0.43/job/submit
```

3. Notez l'ID de tâche renvoyé dans la réponse à des fins de surveillance.

Pour surveiller l'état du travail

1. Obtenez des informations sur un poste spécifique :

```
curl -X GET -H "Authorization: Bearer <jwt>" \
    https://<privateIpAddress>:6820/slurm/v0.0.43/job/<job-id>
```

2. Répertoriez toutes les tâches pour l'utilisateur authentifié :

```
curl -X GET -H "Authorization: Bearer <jwt>" \
    https://<privateIpAddress>:6820/slurm/v0.0.43/jobs
```

Pour annuler une tâche

- Envoyez une demande DELETE pour annuler une tâche spécifique :

```
curl -X DELETE -H "Authorization: Bearer <jwt>" \
    https://<privateIpAddress>:6820/slurm/v0.0.43/job/<job-id>
```

Questions fréquemment posées sur l'API REST de Slurm sur PCS AWS

Cette section répond aux questions fréquemment posées sur l'API REST de Slurm dans AWS PCS.

Qu'est-ce que l'API REST de Slurm ?

L'API REST de Slurm est une interface HTTP qui vous permet d'interagir avec le gestionnaire de charge de travail Slurm par programmation. Vous pouvez utiliser des méthodes HTTP standard telles que GET, POST et DELETE pour soumettre des tâches, surveiller l'état du cluster et gérer les ressources sans avoir besoin d'un accès par ligne de commande au cluster.

Puis-je utiliser des jetons générés par **`scontrol token`** ?

Non, la `scontrol token` sortie standard n'est pas compatible avec le AWS PCS. L'API REST PCS Slurm nécessite des jetons JWT enrichis contenant des revendications d'identité spécifiques, notamment le nom d'utilisateur (`sun`), l'ID utilisateur POSIX (`uid`) et le groupe (). IDs `gids` Les jetons Slurm standard ne présentent pas ces réclamations requises et seront rejetés par l'API.

Puis-je accéder à l'API depuis l'extérieur de mon VPC ?

Non, le point de terminaison de l'API REST n'est accessible que depuis votre VPC à l'aide de l'adresse IP privée du contrôleur Slurm. Pour activer l'accès externe, implémentez AWS des services tels que Application Load Balancer avec VPC Link, API Gateway, ou établissez un peering VPC ou des connexions VPN pour une connectivité sécurisée.

Pourquoi l'API utilise-t-elle le protocole HTTP au lieu du protocole HTTPS ?

L'API REST de Slurm est destinée à être un point de terminaison interne au sein du réseau privé de votre cluster. Pour les déploiements de production nécessitant un chiffrement, vous pouvez implémenter la SSL/TLS terminaison à un niveau supérieur dans votre architecture, par exemple via une passerelle API, un équilibreur de charge ou un proxy inverse.

Comment contrôler l'accès à l'API REST ?

Configurez les règles du groupe de sécurité de votre cluster pour restreindre l'accès au port 6820 sur le contrôleur Slurm. Définissez des règles entrantes pour autoriser les connexions uniquement à partir de plages d'adresses IP fiables ou de sources spécifiques au sein de votre VPC, bloquant ainsi tout accès non autorisé au point de terminaison de l'API.

Comment puis-je faire pivoter la clé de signature JWT ?

Mettez votre cluster en mode maintenance sans aucune instance active, puis lancez la rotation des clés via AWS Secrets Manager. Une fois la rotation terminée, réactivez les files d'attente. Tous les jetons JWT existants deviendront invalides et devront être régénérés à l'aide de la nouvelle clé de signature de Secrets Manager.

Dois-je activer la comptabilité Slurm pour utiliser l'API REST ?

Non, la comptabilité Slurm n'est pas requise pour les opérations de base de l'API REST telles que la soumission et la surveillance des tâches. Cependant, l'ensemble du point de `/slurmdb` terminaison nécessite que la comptabilité soit active.

Quels outils tiers fonctionnent avec l'API REST du AWS PCS ?

De nombreux clients d'API REST Slurm existants devraient fonctionner avec des AWS PCS, notamment Slurm Exporter pour Prometheus, et des applications personnalisées qui suivent le format standard de l'API REST de Slurm. SlurmWeb Cependant, les outils qui reposent sur l'authentification devront être modifiés `scontrol token` pour fonctionner avec les exigences du AWS PCS JWT.

L'utilisation de l'API REST entraîne-t-elle des coûts supplémentaires ?

Non, l'activation ou l'utilisation de la fonctionnalité API REST de Slurm sont gratuites. Vous ne payez que pour les ressources du cluster sous-jacent comme d'habitude.

Comment puis-je résoudre les problèmes liés à l'API REST ?

- Problèmes de connectivité réseau

  Si vous ne pouvez pas atteindre le point de terminaison de l'API, des délais de connexion ou des erreurs de « connexion refusée » s'afficheront lorsque vous envoyez des requêtes HTTP au contrôleur de cluster.

  Que faire : Vérifiez que votre client se trouve dans le même VPC ou dispose d'un routage réseau approprié, et vérifiez que votre groupe de sécurité autorise le trafic HTTP sur le port 6820 à partir de votre adresse IP source ou de votre sous-réseau.

- Problèmes d'authentification REST avec Slurm

  Si votre jeton JWT n'est pas valide, a expiré ou n'est pas correctement signé, les demandes d'API renverront « Erreur d'authentification du protocole » dans le champ des erreurs de la réponse.

  Exemple de message d'erreur :

  ```
  {
  "errors": [
      {
      "description": "Batch job submission failed",
      "error_number": 1007,
      "error": "Protocol authentication error",
      "source": "slurm_submit_batch_job()"
      }
    ]
  }
  ```

  Que faire : Vérifiez que votre jeton JWT est correctement formaté, qu'il n'a pas expiré et qu'il est signé avec la bonne clé dans Secrets Manager. Vérifiez que le jeton est correctement formé, qu'il inclut les revendications requises et que vous utilisez le format d'en-tête d'authentification correct.

- Le Job ne s'exécute pas après sa soumission

  Si votre jeton JWT est valide mais contient une structure ou un contenu interne incorrect, les tâches sont peut-être entrées dans un état suspendu (PD) avec un code de raison. `JobAdminHead` `scontrol show job` *`<job-id>`*À utiliser pour inspecter le travail, vous verrez`JobState=PENDING, Reason=JobHeldAdmin`, et`SystemComment=slurm_cred_create failure, holding job`.

  Que faire : La cause première peut être des valeurs erronées dans JWT. Vérifiez que le jeton est correctement structuré et inclut les allégations requises conformément à la documentation du PCS.

- Problèmes d'autorisation liés au répertoire de travail

  Si l'identité utilisateur spécifiée dans votre JWT ne dispose pas d'autorisations d'écriture dans le répertoire de travail de la tâche, celle-ci échouera avec des erreurs d'autorisation, comme si vous `sbatch --chdir` utilisiez un répertoire inaccessible.

Que faire : assurez-vous que l'utilisateur spécifié dans votre jeton JWT dispose des autorisations appropriées pour le répertoire de travail de la tâche.

Redémarrage de nœuds de calcul avec Slurm sur PCS AWS

AWS PCS supporte la commande native `scontrol reboot` de Slurm. Utilisez cette commande pour redémarrer les nœuds de calcul sans remplacer l'instance EC2. D'autres méthodes de redémarrage (console Amazon EC2 AWS CLI, correctifs automatisés ou maintenance du système) amènent les AWS PC à considérer que l'instance EC2 est défectueuse et à la remplacer.

Avantages du redémarrage de Slurm

Le redémarrage de Slurm présente plusieurs avantages pour la maintenance des clusters :

- Préservez la capacité : évitez de perdre des instances EC2 à capacité limitée au profit d'autres clients.
- Réduisez les coûts : éliminez les cycles de remplacement d'instances inutiles et la facturation continue pour les nœuds inactifs.
- Restauration plus rapide : aucun retard de provisionnement par rapport au remplacement d'une instance.
- Flexibilité opérationnelle — Éliminez les fuites de mémoire, supprimez les fichiers temporaires et restaurez les nœuds en état de dégradation.

Quand utiliser le redémarrage de Slurm

Utilisez le redémarrage de Slurm pour les scénarios de maintenance opérationnelle courants :

- Dépannage — Résolvez les problèmes de performances ou les processus qui ne répondent pas, en particulier pour les nœuds GPU.
- Nettoyage des ressources : éliminez les fuites de mémoire, les fichiers temporaires ou les `/tmp` processus bloqués qui affectent les performances au travail.
- Restauration : restaurez les nœuds à l'état bloqué ou dégradé avant de demander leur remplacement complet.

Limitations

- Seuls les utilisateurs Slurm Admin (utilisateurs root) peuvent exécuter des commandes de redémarrage.
- Le support de redémarrage est `scontrol reboot` limité à.
- RebootProgram la configuration n'est pas prise en charge.
- Aucune interface de console, ligne de commande uniquement.

Rubriques

- Redémarrer un nœud de calcul à l'aide de Slurm dans PCS AWS
- Annuler un redémarrage en attente dans AWS PCS
- Questions fréquemment posées sur le redémarrage de Slurm sur PCS AWS
- Résolution des problèmes de redémarrage de Slurm sur PCS AWS

Redémarrer un nœud de calcul à l'aide de Slurm dans PCS AWS

Utilisez la commande de redémarrage native de Slurm pour résoudre les problèmes de performances, résoudre les problèmes de ressources ou récupérer après un état dégradé sans perte de capacité de l'instance EC2.

Conditions préalables

- Privilèges d'administrateur Slurm (accès utilisateur root)
- Accès à un nœud de connexion dans le cluster AWS PCS

Procédure

1. Connectez-vous à un nœud de connexion via la console EC2.
    a. Dans la console EC2, choisissez Instances.
    b. Sélectionnez votre instance de nœud de connexion.
    c. Choisissez Se connecter.
2. Identifiez le nom du nœud de calcul cible à l'aide de `sinfo` ou`scontrol show node`.

```
sinfo
```

```
# or
scontrol show node
```

3. Exécutez la commande de redémarrage à l'aide de l'une des options suivantes :

> **Warning**
>
> Ne l'utilisez pas `nextstate=DOWN` avec la `scontrol reboot` commande. Ce paramètre indique que le nœud est défectueux et déclenche le remplacement de l'instance.

- Redémarrage de base (attend que le nœud soit inactif) :

```
scontrol reboot nodename
```

- Redémarrage immédiat (vide le nœud et redémarre une fois les tâches terminées) :

```
scontrol reboot ASAP nodename
```

- Redémarrez avec raison :

```
scontrol reboot ASAP reason="troubleshooting" nodename
```

- Redémarrez avec l'état de reprise :

```
scontrol reboot ASAP nextstate=RESUME nodename
```

4. Surveillez la progression du redémarrage à l'aide de`scontrol show node`.

```
scontrol show node nodename
```

5. Vérifiez que le nœud est remis en service une fois le redémarrage terminé.

Annuler un redémarrage en attente dans AWS PCS

Annulez un redémarrage en attente pour éviter toute interruption inutile une fois le problème résolu ou lorsque le redémarrage n'est plus nécessaire.

Conditions préalables

- Privilèges d'administrateur de Slurm
- Le nœud doit avoir un redémarrage en attente (indiquant le statut « redémarrage émis »)
- Accès au nœud de connexion pour l'exécution des commandes

Procédure

1. Connectez-vous au nœud de connexion.
2. Vérifiez que le nœud est en attente de redémarrage à l'aide de`scontrol show node`.

   ```
   scontrol show node nodename
   ```

   Recherchez « problème de redémarrage » dans l'état du nœud.
3. Exécutez la commande d'annulation.

   ```
   scontrol cancel_reboot nodename
   ```
4. Vérifiez l'annulation du redémarrage et le retour à la normale de l'état du nœud.

   ```
   scontrol show node nodename
   ```

Questions fréquemment posées sur le redémarrage de Slurm sur PCS AWS

Trouvez les réponses aux questions les plus fréquemment posées sur l'utilisation de Slurm reboot sur PCS. AWS

Qu'est-ce que le support de redémarrage de Slurm ?

Support de la commande native Slurm. `scontrol reboot` Utilisez cette commande pour redémarrer les nœuds de calcul sans remplacement automatique des instances, afin de préserver la capacité des instances EC2 et de réduire les coûts opérationnels.

Qui peut utiliser les commandes de redémarrage de Slurm ?

Seuls les utilisateurs Slurm Admin (utilisateurs root) peuvent exécuter des commandes de redémarrage. Les utilisateurs réguliers qui tentent de l'utiliser `scontrol reboot` recevront une erreur de refus d'autorisation de la part de Slurm sans que cela n'affecte le nœud.

Qu'arrive-t-il à l'exécution de tâches lors d'un redémarrage ?

Par défaut, les tâches se terminent normalement avant le redémarrage. Avec l'option ASAP, le nœud est vidé pour empêcher de nouvelles tâches, et le redémarrage a lieu une fois les tâches en cours terminées. Les tâches peuvent être annulées ou mises en attente pour un redémarrage immédiat.

En quoi est-ce différent du redémarrage de la console EC2 ?

Le redémarrage lent préserve l'instance EC2 et évite son remplacement, tandis que le redémarrage de la console EC2 incite PCS à remplacer l'instance en raison de l'échec des contrôles de santé effectués lors du processus de redémarrage.

Puis-je configurer des scripts de redémarrage personnalisés ?

Non, RebootProgram la configuration n'est pas prise en charge dans la version initiale. Cette fonctionnalité utilise le comportement de redémarrage standard de Slurm sans prise en charge de scripts personnalisés.

Combien de temps dure un redémarrage de Slurm ?

Le temps de redémarrage varie en fonction du type d'instance, des processus de démarrage du client, de la configuration de l'AMI et de la nécessité ou non de terminer les tâches en premier. Le processus inclut l'attente de la fin des tâches, le redémarrage physique, les vérifications de santé et l'enregistrement du daemon slurmd.

Puis-je consulter l'historique des redémarrages ?

Les événements de redémarrage sont enregistrés dans les journaux de Slurm (slurmctld et slurmd) qui peuvent être surveillés par le biais de ce journal. CloudWatch Le champ Motif dans l'état du nœud indique le motif du redémarrage au cours du processus.

Que faire si un nœud est bloqué pendant le redémarrage ?

Si un nœud ne termine pas le processus de redémarrage qu'il ResumeTimeout contient, il sera marqué comme ÉTANT HORS SERVICE. Vérifiez la présence d'erreurs dans les CloudWatch journaux, vérifiez la connectivité réseau et examinez les journaux slurmd. Contactez AWS le Support si les problèmes persistent.

Puis-je redémarrer plusieurs nœuds à la fois ?

Oui, vous pouvez spécifier plusieurs nœuds dans la commande de redémarrage :

```
scontrol reboot ASAP node1,node2,node3
```

Comment puis-je redémarrer un nœud sans attendre la fin des tâches ?

Pour un redémarrage immédiat des nœuds en cas de problèmes tels que des nœuds problématiques affectant des tâches multi-nœuds, une dégradation significative des performances ou un comportement instable du GPU, deux options s'offrent à vous :

- Annuler et redémarrer — Tout d'abord, annulez les tâches concernées à l'aide de`scancel <job_id>`, puis lancez un redémarrage immédiat à l'aide de`scontrol reboot ASAP <nodename>`. Les tâches en cours seront interrompues et devront être soumises à nouveau une fois le nœud rétabli.
- Vidange et mise en file d'attente (moins d'impact) : commencez par lancer une vidange et redémarrez avec`scontrol reboot ASAP <nodename>`, puis mettez en file d'attente les tâches concernées en utilisant. `scontrol requeue <job_id>` Cela permet de remettre les emplois en attente au lieu de les annuler.

Que se passe-t-il si je spécifie NextState=DOWN ?

Si vous le spécifiez`nextstate=DOWN`, le nœud sera marqué comme défectueux après le redémarrage et déclenchera le remplacement de l'instance. Pour éviter le remplacement d'instance, ne spécifiez pas nextstate ou use`nextstate=RESUME`.

Ressources supplémentaires

- Pour les procédures de redémarrage de base, voirRedémarrer un nœud de calcul à l'aide de Slurm dans PCS AWS.
- Pour résoudre les problèmes de redémarrage, voirRésolution des problèmes de redémarrage de Slurm sur PCS AWS.
- Pour la documentation sur le redémarrage de Slurm, consultez la documentation de Slurm scontrol.

Résolution des problèmes de redémarrage de Slurm sur PCS AWS

Lorsque vous rencontrez des problèmes de redémarrage d'un nœud, vérifiez d'abord l'état du nœud à l'aide de`scontrol show node` *`nodename`*. Examinez ensuite les CloudWatch journaux de Slurm (slurmctld et slurmd) et les journaux du système afin d'identifier les erreurs potentielles.

Pour le dépannage de base, vérifiez la connectivité réseau, vérifiez les paramètres du groupe de sécurité et assurez-vous que tous les services requis fonctionnent après le redémarrage. Si les problèmes persistent après les étapes de dépannage de base, contactez AWS le Support. Lorsque

vous contactez le support, fournissez des extraits de journal pertinents, des informations sur l'état du nœud et une chronologie de la tentative de redémarrage afin d'accélérer le processus de résolution.

Ressources supplémentaires

- Pour surveiller les instances AWS PCS à l'aide d'Amazon CloudWatch, consultez la section Surveillance des instances AWS PCS à l'aide d'Amazon CloudWatch.
- Pour un dépannage général, voirRésolution des problèmes dans le service de calcul AWS parallèle.
- Pour la documentation de Slurm, consultez le guide de résolution des problèmes de Slurm.

Configuration des paramètres personnalisés de Slurm dans PCS AWS

Utilisez des paramètres Slurm personnalisés pour configurer des paramètres Slurm supplémentaires sur les ressources du cluster, de la file d'attente et du groupe de nœuds de calcul. Cette version ajoute la prise en charge des paramètres Slurm sur les ressources de file d'attente, offrant ainsi un contrôle granulaire des comportements spécifiques aux partitions.

Avantages des paramètres personnalisés de Slurm

Les paramètres personnalisés de Slurm fournissent un contrôle sophistiqué de votre environnement AWS HPC basé sur PC. Vous pouvez mettre en œuvre une comptabilité détaillée, appliquer des contrôles d'accès et optimiser l'exécution de la charge de travail grâce à des quality-of-service configurations et à des politiques de préemption. Ces fonctionnalités garantissent que les tâches critiques reçoivent les ressources nécessaires tout en maintenant une utilisation efficace du cluster. Que vous gériez des charges de travail accélérées par le GPU, que vous implémentiez une planification équitable ou que vous contrôliez le cycle de vie des tâches, les paramètres personnalisés vous aident à aligner votre infrastructure HPC sur les exigences opérationnelles et les objectifs de recherche.

Configuration de paramètres personnalisés

Les paramètres personnalisés de Slurm peuvent être configurés via la AWS console, la CLI, SDKs lors de la création de ressources ou modifiés ultérieurement par le biais d'opérations de mise à jour.

AWS Management Console

Accédez aux paramètres supplémentaires du planificateur dans la page de création ou de modification pour tout type de ressource (cluster, file d'attente ou groupe de nœuds de calcul).

Pour ajouter un nouveau paramètre

1. Choisissez Ajouter un nouveau paramètre.
2. Sélectionnez un nom de paramètre dans la liste déroulante (qui inclut de brèves descriptions des paramètres).
3. Indiquez la valeur correspondante.

Pour annuler un paramètre personnalisé

1. Choisissez Supprimer à côté de la parameter/value paire correspondante.
2. Créez ou mettez à jour la ressource.

AWS CLI

Pour la gestion programmatique des paramètres personnalisés, utilisez le `SlurmCustomSettings` champ dans les opérations de création ou de mise à jour.

Example— Mise à jour du Prolog paramètre sur un cluster

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

Example— Configuration d'une file d'attente pour qu'elle Default figure sur un cluster

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

Example— Configuration personnalisée Features sur un groupe de nœuds de calcul

```
aws pcs update-compute-node-group \
```

```
    --cluster-identifier my-cluster \
    --compute-node-group-identifier my-cng-1 \
    --slurm-configuration \
    'SlurmCustomSettings=[{parameterName=Features,parameterValue="gpu,nvme"}]'
```

Validation et gestion des erreurs

AWS PCS met en œuvre un processus de validation à plusieurs niveaux pour les paramètres personnalisés de Slurm. Au cours des opérations de création et de mise à jour, nous effectuons des validations synchrones qui incluent :

- Contrôles au niveau du terrain : nous validons les paramètres individuels pour les types de données corrects, les valeurs autorisées et les exigences de format. Par exemple, nous nous assurons que les valeurs temporelles sont au format Slurm correct et que les valeurs booléennes utilisent les représentations booléennes Slurm acceptées.
- Validations sensibles au contexte : certains paramètres sont vérifiés par rapport au contexte de configuration plus large. Par exemple, certains paramètres ne sont valides que lorsque la comptabilité Slurm est activée.
- Cohérence entre les paramètres : nous vérifions que les options qui s'excluent mutuellement ne sont pas définies ensemble et que les paramètres interdépendants sont correctement configurés.

Si la validation échoue, vous recevrez un `ValidationException` code d'erreur spécifique (par exemple, InvalidInput), un message d'erreur clair décrivant le problème, ainsi qu'une liste des champs non valides et leurs informations d'erreur respectives.

Bien que de nombreux problèmes soient détectés lors de cette validation initiale, certaines interactions complexes entre les paramètres ne peuvent apparaître que lors de l'application de la configuration. Dans ce cas, l'opération échouera avec un message d'erreur informatif, et toute modification partielle sera annulée.

Limitations

AWS Le PCS met en œuvre une approche de liste d'autorisation pour protéger la sécurité des services et la stabilité opérationnelle. Les paramètres susceptibles de compromettre la sécurité des comptes de service ou d'interférer avec les fonctionnalités des services gérés sont limités. Cependant, nous évaluons en permanence les besoins des clients et pouvons ajouter un support pour des paramètres supplémentaires en fonction des commentaires des clients.

Rubriques

- Paramètres Slurm personnalisés pour les clusters PCS AWS
- Paramètres Slurm personnalisés pour les groupes de nœuds de calcul AWS PCS
- Paramètres Slurm personnalisés pour AWS les files d'attente PCS
- Résolution des problèmes liés aux paramètres personnalisés de Slurm dans PCS AWS

Paramètres Slurm personnalisés pour les clusters PCS AWS

Les paramètres Slurm personnalisés suivants sont pris en charge au niveau du cluster :

- AccountingStorageEnforce

> **Important**
>
> AWS PCS prend en charge un sous-ensemble des options pour`AccountingStorageEnforce`. Pour de plus amples informations, veuillez consulter Comptabilité Slurm dans PCS AWS.

- AccountingStorageTRES
- AccountingStoreFlags
- DefMemPerCPU
- Epilog
- EnforcePartLimits
- FairShareDampeningFactor
- HealthCheckInterval
- HealthCheckNodeState
- HealthCheckProgram
- JobRequeue
- LaunchParameters
- Licenses
- MinJobAge

> **Note**
>
> AWS PCS prend en charge une valeur minimale de 5 secondes pour`MinJobAge`.

- OverTimeLimit
- PreemptExemptTime
- PreemptMode
- PreemptParameters
- PreemptType
- PriorityCalcPeriod
- PriorityDecayHalfLife
- PriorityFavorSmall
- PriorityFlags
- PriorityMaxAge
- PriorityUsageResetPeriod
- PriorityWeightAge
- PriorityWeightAssoc
- PriorityWeightFairshare
- PriorityWeightJobSize
- PriorityWeightPartition
- PriorityWeightQOS
- PriorityWeightTRES
- PrivateData
- Prolog
- PrologFlags
- PropagatePrioProcess
- PropagateResourceLimits
- PropagateResourceLimitsExcept
- RequeueExit

- RequeueExitHold
- SchedulerParameters
- SelectTypeParameters
- SrunPortRange
- TaskEpilog
- TaskPluginParam
- TaskProlog
- UnkillableStepProgram
- UnkillableStepTimeout

Paramètres Slurm personnalisés pour les groupes de nœuds de calcul AWS PCS

Les paramètres Slurm personnalisés suivants sont pris en charge au niveau du groupe de nœuds de calcul :

- CpuSpecList
- Features
- MemSpecLimit
- RealMemory
- Weight

Paramètres Slurm personnalisés pour AWS les files d'attente PCS

Les paramètres Slurm personnalisés suivants sont pris en charge au niveau de la file d'attente :

- AllowAccounts
- AllowQoS
- Default
- DefaultTime
- DenyAccounts
- DenyQoS

- ExclusiveUser
- GraceTime
- MaxTime
- OverSubscribe
- OverTimeLimit
- PreemptMode
- PriorityJobFactor
- PriorityTier
- QOS
- TRESBillingWeights

Résolution des problèmes liés aux paramètres personnalisés de Slurm dans PCS AWS

Si vous rencontrez des erreurs lors de la création ou de la mise à jour des ressources AWS PCS avec les paramètres personnalisés de Slurm, vous pouvez utiliser la journalisation pour diagnostiquer et résoudre les problèmes.

Résolution des problèmes liés aux paramètres personnalisés incompatibles de Slurm

Problème : vous recevez un message d'erreur similaire au suivant lorsque vous effectuez des opérations de cluster, de groupe de nœuds de calcul ou de file d'attente :

```
{OPERATION} failed. The Slurm custom settings of the cluster might be incompatible.
 Check the settings and try again.
```

Cette erreur peut se produire lors des opérations suivantes :

- CreateCluster
- CreateComputeNodeGroup
- UpdateComputeNodeGroup
- CreateQueue
- UpdateQueue

Solution : Activez la journalisation pour comprendre le problème spécifique et résoudre les problèmes liés aux paramètres incompatibles.

Pour résoudre les problèmes liés aux paramètres personnalisés incompatibles de Slurm

1. Créez le cluster s'il n'existe pas encore, ou assurez-vous que votre cluster existant est dans un état dans lequel la journalisation peut être activée.
2. Activez la journalisation pour votre cluster. Pour obtenir des instructions complètes, consultez Journalisation et surveillance pour AWS PCS.

   > **Note**
   >
   > La journalisation peut être activée une fois que le cluster est en cours de création.

3. Consultez les journaux pour identifier le problème de configuration spécifique de Slurm à l'origine de l'incompatibilité.
4. Corrigez les paramètres personnalisés incompatibles en fonction des informations du journal et recommencez l'opération.

Pour plus d'informations sur les paramètres personnalisés pris en charge par Slurm, voir :

- Paramètres Slurm personnalisés pour les clusters PCS AWS
- Paramètres Slurm personnalisés pour les groupes de nœuds de calcul AWS PCS
- Paramètres Slurm personnalisés pour AWS les files d'attente PCS

Étendez les fonctionnalités de Slurm sur AWS PC avec les plugins SPANK

Utilisez les plugins SPANK (Slurm Plug-in Architecture for Node and job Kontrol) pour étendre et modifier le comportement de Slurm lors du lancement et de l'exécution des tâches sur des clusters PCS. AWS Les plugins SPANK fournissent une interface générique pour intercepter et modifier les étapes de lancement des tâches.

Installez les plugins SPANK sur l'AMI de votre nœud de calcul et configurez-les pour personnaliser le comportement de votre cluster Slurm en fonction des exigences de votre charge de travail. Pour plus d'informations sur SPANK, consultez la documentation SPANK sur le site Web de SchedMD.

Table des matières

- Installez les plugins SPANK sur PC AWS
- Configurer les plugins SPANK sur PCS AWS
- Questions fréquemment posées à propos des plugins SPANK sur PC AWS

Installez les plugins SPANK sur PC AWS

Suivez la documentation du plugin pour installer les plugins SPANK sur votre AMI.

Compilez les plugins SPANK pour la version spécifique de Slurm sur votre cluster. Le programme d'installation de Slurm fourni par AWS PCS stocke Slurm dans. `/opt/aws/pcs/scheduler/slurm-version` Lorsque vous compilez le plugin, spécifiez la version de Slurm.

L'exemple suivant montre comment spécifier la version de Slurm pour certains plugins :

```
export CFLAGS="-I/opt/aws/pcs/scheduler/slurm-version/include"
```

Si vous avez plusieurs versions de Slurm dans l'AMI, compilez le plugin pour chaque version. Stockez les plugins compilés dans des dossiers versionnés.

L'exemple suivant montre comment spécifier le dossier de destination pour certains plugins :

```
export DESTDIR="your-preferred-versioned-path"
```

> **Important**
>
> Les plugins peuvent nécessiter des variables différentes. Consultez la documentation officielle du plugin que vous installez.

Configurer les plugins SPANK sur PCS AWS

Par défaut, stockez les fichiers de configuration dans`/etc/aws/pcs/scheduler/slurm-version/plugstack.conf.d/`.

Pour stocker votre configuration SPANK dans un autre emplacement, ajoutez vos emplacements à un fichier de configuration dans le répertoire par défaut.

L'exemple suivant montre comment inclure des fichiers de configuration provenant d'autres répertoires :

```
# content of /etc/aws/pcs/scheduler/slurm-version/any-filename.conf
include path-to-your-configuration-folder/*.conf
include path-to-a-second-configuration-folder/*.conf
```

Stockez chaque configuration dans un fichier dédié ou dans un fichier commun. Vous pouvez utiliser plusieurs fichiers de configuration.

Les exemples suivants présentent des exemples de fichiers de configuration :

```
# content of path-to-your-or-default-config-folder/filename-1.conf
required path-to-plugin-1 arguments
optional path-to-plugin-2 arguments
```

```
# content of path-to-your-or-default-config-folder/filename-2.conf
required path-to-plugin-3 arguments
```

Pour plus d'informations sur la configuration de vos plugins, consultez la documentation de configuration de SPANK sur le site Web de SchedMD.

Important

Définissez des autorisations de dossier pour empêcher toute modification non autorisée de la configuration de votre plugin.

Note

AWS PCS ne gère pas vos plugins SPANK. Si vous rencontrez des erreurs liées aux plugins, consultez les journaux d'erreurs de vos nœuds de calcul.

Note

Slurm enregistre incorrectement une erreur similaire à la suivante lorsqu'il charge votre configuration SPANK :

```
error: "Include" failed in file /etc/slurm/plugstack.conf line 3
```

Vous pouvez ignorer cette erreur. Cela n'affecte pas le fonctionnement des plugins SPANK.

Questions fréquemment posées à propos des plugins SPANK sur PC AWS

Cette section aborde les questions courantes concernant l'installation et la configuration des plugins SPANK sur les clusters AWS PCS.

Dois-je installer les plugins SPANK à la fois sur les nœuds de connexion et les nœuds de calcul ?

Certains plugins SPANK ne nécessitent pas d'installation sur tous les nœuds, mais pour une meilleure compatibilité, nous vous recommandons d'installer tous les plugins SPANK sur chaque nœud.

Quelle configuration supplémentaire est nécessaire pour l'utilisation en production des plugins SPANK ?

Au-delà de l'installation et de la configuration de base présentées dans les exemples, les déploiements de production nécessitent généralement une configuration supplémentaire. Les plugins basés sur des conteneurs tels que Pyxis peuvent vous obliger à définir des variables d'environnement pour Enroot, à activer le PMI (Process Management Interface) et à configurer des autorisations pour l'exécution du conteneur. Consultez la documentation du plugin spécifique pour connaître les exigences de déploiement détaillées en production.

Comment résoudre les problèmes liés au plugin SPANK ?

AWS PCS ne gère pas les plugins SPANK. Consultez les journaux d'erreurs de vos nœuds de calcul pour résoudre les problèmes.

Utilisez les plugins de filtre Slurm CLI pour personnaliser la soumission des tâches dans PCS AWS

AWS PCS prend en charge les plugins de filtre CLI Slurm pour exécuter des scripts Lua personnalisés qui valident et modifient les paramètres de soumission des tâches sur les nœuds de connexion et de calcul. Pour des informations détaillées sur les plug-ins CLI Filter, consultez la documentation de l'API du plug-in cli_filter sur le site Web de SchedMD.

Exigences

Les plugins CLI Filter nécessitent la version 24.11 ou ultérieure de Slurm et un script Lua déployé sur tous les nœuds de connexion et de calcul.

> **Important**
>
> Pour les versions 24.11 et 25.05 de Slurm, les plugins CLI Filter nécessitent l'installation de Slurm à l' AWS aide du programme d'installation de PCS Slurm (version 24.11.6-2+ ou 25.05.4-1+). Pour plus d'informations sur l'installation de Slurm, consultez. Étape 3 — Installation de Slurm

Limites et considérations de sécurité

- Application de la sécurité — Les plug-ins de filtre CLI peuvent être facilement contournés par n'importe quel utilisateur et ne doivent pas être utilisés pour des politiques de sécurité critiques. Les utilisateurs peuvent désactiver les plug-ins de filtrage CLI en fournissant une configuration personnalisée qui a été `CLIFilterPlugins` désactivée lors de la soumission des tâches.
- Implémentation de Lua uniquement : l'implémentation de scripts Lua est prise en charge. L'implémentation en C n'est pas prise en charge.

Rubriques

- Configuration des plugins de filtre CLI Slurm sur un cluster PCS AWS
- Utiliser Amazon S3 pour déployer un script de plug-in de filtrage CLI dans AWS PCS
- Translate un script du plugin Slurm Job Submit pour utiliser le plugin CLI Filter dans PCS AWS
- Questions fréquemment posées sur les plug-ins de filtre Slurm CLI dans PCS AWS
- Résolution des problèmes liés au plugin Slurm CLI Filter sur PCS AWS

Configuration des plugins de filtre CLI Slurm sur un cluster PCS AWS

Configurez les plug-ins de filtre CLI lorsque vous créez un nouveau cluster AWS PCS. Vous pouvez activer ou désactiver les plug-ins de filtrage CLI sur les clusters existants à l'aide de l'API de mise à jour ou de la console sans recréer le cluster.

Conditions préalables

Avant de configurer les plug-ins de filtre CLI, effectuez les tâches suivantes :

- Écrire et tester un script Lua qui implémente l'API du plugin CLI Filter
- Nommez votre script Lua exactement `cli_filter.lua`
- Choisissez une méthode pour déployer votre script sur toutes les instances de cluster (AMI, S3 ou système de fichiers)
- Vérifiez que vous utilisez la version 24.11 ou ultérieure de Slurm

Activer les plug-ins de filtrage CLI sur un nouveau cluster

AWS PCS console

1. Ouvrez la console AWS PCS à l'adresse https://console.aws.amazon.com/pcs/.
2. Dans le panneau de navigation, choisissez Clusters.
3. Choisissez Créer un cluster.
4. Sélectionnez une version valide de Slurm (version 24.11 ou ultérieure).
5. Sous Paramètres du planificateur, développez Paramètres supplémentaires du planificateur.
6. Ajoutez un nouveau paramètre personnalisé Slurm avec le nom du paramètre défini sur `CliFilterPlugins` et la valeur du paramètre définie sur. `cli_filter/lua`
7. Terminez la configuration de cluster restante et choisissez Create cluster.

AWS PCS API

Fournissez la `slurmCustomSettings` configuration dans votre appel à l'action d'`CreateCluster`API. Réglez le `parameterName` point `CliFilterPlugins` et `parameterValue` le point`cli_filter/lua`. Pour plus d'informations, consultez CreateClusterla référence de l'API AWS PCS.

L'exemple suivant utilise le AWS CLI pour appeler l'action d'`CreateCluster`API. Le paramètre personnalisé `CliFilterPlugins=cli_filter/lua` active les plug-ins de filtre CLI.

```
aws pcs create-cluster --cluster-name cluster-name \
--scheduler type=SLURM,version=24.11 \
```

```
--size SMALL \
--networking subnetIds=cluster-subnet-id,securityGroupIds=cluster-security-group-id
 \
--slurm-configuration \
'slurmCustomSettings=[{parameterName=CliFilterPlugins,parameterValue="cli_filter/
lua"}]'
```

Déployer les scripts du plugin CLI Filter

Pour déployer des scripts du plug-in CLI Filter sur votre cluster

1. Assurez-vous que Slurm AMIs est installé sur tous les groupes de nœuds de calcul via le programme d'installation de AWS PCS Slurm.

   > **Note**
   >
   > Si vous utilisez l'AMI AWS PCS Sample pour tous les groupes de nœuds de calcul, ignorez cette étape. Slurm est déjà installé.

2. Déployez votre `cli_filter.lua` script `/etc/aws/pcs/scheduler/slurm-<version>/cli_filter.lua` sur toutes les instances du cluster.

   Par exemple, pour la version 24.11 de Slurm :

   ```
   /etc/aws/pcs/scheduler/slurm-24.11/cli_filter.lua
   ```

3. Lancez tous les nœuds de connexion et de calcul à l'aide de votre système préparé AMIs.
4. Testez la soumission des tâches pour vérifier que le plug-in CLI Filter s'exécute correctement.

Activer ou désactiver les plug-ins de filtrage CLI sur les clusters existants

Vous pouvez activer ou désactiver les plug-ins de filtrage CLI sur les clusters existants sans avoir à reconstruire votre infrastructure. Pour de plus amples informations, veuillez consulter Mettre à jour un cluster dans AWS PCS.

AWS PCS console

1. Ouvrez la console AWS PCS à l'adresse https://console.aws.amazon.com/pcs/.
2. Dans le panneau de navigation, choisissez Clusters.

3. Sélectionnez le cluster à mettre à jour.
4. Choisissez l'action Modifier.
5. Sur la page Modifier le cluster, sous Paramètres supplémentaires du planificateur :
   - Pour activer les plug-ins de filtrage CLI : ajoutez un nouveau paramètre personnalisé Slurm avec le nom du paramètre défini sur `CliFilterPlugins` et la valeur du paramètre définie sur. `cli_filter/lua`
   - Pour désactiver les plug-ins de filtrage CLI : supprimez le `CliFilterPlugins` paramètre existant.
6. Choisissez Mettre à jour le cluster pour soumettre les modifications.
7. Surveillez l'état du cluster, qui s'affiche comme « Mise à jour » pendant le processus et « Actif » lorsque la mise à jour est terminée.

AWS PCS API

Utilisez l'action `UpdateCluster` API pour activer ou désactiver les plug-ins de filtre CLI. Pour plus d'informations, consultez UpdateClusterla référence de l'API AWS PCS.

Pour activer les plug-ins de filtrage CLI sur un cluster existant :

```
aws pcs update-cluster --cluster-identifier my-cluster \
--slurm-configuration \
'slurmCustomSettings=[{parameterName=CliFilterPlugins,parameterValue="cli_filter/
lua"}]'
```

Pour désactiver les plug-ins de filtrage CLI sur un cluster existant, procédez comme suit :

```
aws pcs update-cluster --cluster-identifier my-cluster \
--slurm-configuration \
'slurmCustomSettings=[]'
```

Résultats attendus

Après avoir terminé la configuration :

- Votre cluster est créé avec le plug-in CLI Filter activé

- Les offres d'emploi déclenchent votre logique de validation personnalisée avant d'atteindre le contrôleur Slurm
- Les tâches non conformes sont rejetées avec vos messages d'erreur personnalisés
- Les tâches conformes se déroulent normalement via le planificateur Slurm

Résolution des problèmes

Script du plugin CLI Filter manquant sur n'importe quel nœud

Symptômes : la soumission du Job échoue immédiatement avec une erreur de chargement du plugin.

Cause probable : le script n'a pas été déployé sur toutes les instances ou le chemin ou le nom du fichier est incorrect.

Solution : Vérifiez que le script existe au bon chemin sur tous les nœuds de connexion et de calcul avec le nom de fichier exact`cli_filter.lua`.

Configuration du plugin de filtre CLI non valide

Symptômes : la création du cluster échoue avec une erreur de validation.

Cause probable : `CliFilterPlugins` le paramètre n'est pas défini sur le `cli_filter/lua` format.

Résolution : utilisez la valeur exacte du paramètre `cli_filter/lua` dans`slurmCustomSettings`.

Utiliser Amazon S3 pour déployer un script de plug-in de filtrage CLI dans AWS PCS

Utilisez S3 pour déployer votre script CLI Filter Plugin lorsque vous souhaitez mettre à jour la logique de soumission des tâches sur un cluster actif sans avoir à le reconstruire AMIs. Cette approche télécharge le script depuis S3 lors du lancement de l'instance à l'aide des données utilisateur.

Conditions préalables

Avant de déployer votre script à l'aide de S3, effectuez les tâches suivantes :

- Créez un compartiment S3 avec le script Lua de votre CLI Filter Plugin
- Configuration du profil d'instance IAM avec accès en lecture au compartiment S3
- Configurer le point de terminaison S3 VPC Gateway pour un accès direct sans Internet
- Préparer le script de données utilisateur à télécharger depuis S3

Pour déployer le script du plug-in CLI Filter à l'aide de S3

1. Téléchargez votre `cli_filter.lua` script dans votre compartiment S3.
2. Configurez votre profil d'instance IAM avec des autorisations de lecture S3 pour le compartiment.
3. Ajoutez du code shell aux données utilisateur de votre modèle de lancement pour télécharger le script :

```
aws s3 cp s3://my-bucket/cli_filter.lua /etc/aws/pcs/scheduler/slurm-24.11/
cli_filter.lua
chmod 644 /etc/aws/pcs/scheduler/slurm-24.11/cli_filter.lua
```

4. Déployez des groupes de nœuds de calcul avec vos modèles de lancement mis à jour.
5. Testez la soumission des tâches pour vérifier le fonctionnement du script.

Résultats attendus

Une fois le déploiement de S3 terminé :

- Le script du plugin CLI Filter est automatiquement téléchargé sur toutes les instances lors du lancement
- Les mises à jour des scripts dans S3 sont répercutées sur les instances nouvellement lancées
- Les politiques de soumission de tâches sont appliquées de manière cohérente dans l'ensemble du cluster

Résolution des problèmes

Accès S3 refusé

Symptômes : le lancement de l'instance échoue ou le script n'a pas été téléchargé.

Cause probable : autorisations IAM ou point de terminaison VPC S3 manquants.

Solution : Vérifiez que le profil d'instance IAM est `s3:GetObject` autorisé et que le point de terminaison VPC S3 est configuré.

Translate un script du plugin Slurm Job Submit pour utiliser le plugin CLI Filter dans PCS AWS

Traduisez votre script Lua Job Submit Plugin existant en CLI Filter Plugin lorsque vous migrez depuis d'autres environnements Slurm. Le processus de traduction implique la mise à jour des noms de fonctions et des modèles d'accès aux champs pour qu'ils fonctionnent avec l'API du plugin CLI Filter.

Conditions préalables

Avant de traduire votre script, effectuez les tâches suivantes :

- Passez en revue le script Lua de votre plugin Job Submit existant
- Comprendre les différences entre Job Submit et le plugin CLI Filter APIs
- Accédez à la documentation du plugin Slurm CLI Filter

Pour traduire le script du plugin Job Submit en plugin CLI Filter

1. Passez en revue les fonctions de script de votre plugin Job Submit existantes (`slurm_job_submit`,`slurm_job_modify`).
2. Identifiez les fonctions équivalentes du plugin CLI Filter :
   - `slurm_job_submit`devient `slurm_cli_pre_submit`
   - Ajouter `slurm_cli_setup_defaults` pour le réglage des paramètres par défaut
   - Ajouter `slurm_cli_post_submit` pour les actions postérieures à la soumission
3. Transformez la logique de validation des tâches `job_desc` des champs en accès aux `options` tableaux :
   - `job_desc.account`devient `options["account"]`
   - `job_desc.partition`devient `options["partition"]`
   - `job_desc.features`devient `options["constraint"]`
4. Mettre à jour la journalisation des appels `slurm.log_user()` de à`slurm.log_error()`.
5. Testez votre script traduit sur un cluster de développement.

6. Déployez sur votre cluster de production en suivant le processus standard de déploiement du plug-in CLI Filter.

Résultats attendus

Après avoir terminé la traduction :

- Votre script traduit fournit une validation de soumission de tâche équivalente
- Les utilisateurs voient des messages d'erreur et des instructions similaires à ceux de votre plugin Job Submit d'origine
- Les politiques de soumission de tâches sont maintenues lors de la migration vers AWS PCS

Résolution des problèmes

Erreurs de traduction de script

Symptômes : les soumissions de Job échouent en raison d'erreurs d'exécution de Lua.

Cause probable : accès incorrect aux champs ou appels de fonction dans le script traduit.

Résolution : Consultez la documentation de l'API du plugin CLI Filter et comparez les mappages de champs entre les interfaces Job Submit et CLI Filter.

Questions fréquemment posées sur les plug-ins de filtre Slurm CLI dans PCS AWS

Consultez ces questions fréquemment posées sur les plug-ins de filtre CLI.

Quelle est la différence entre le plugin CLI Filter et le plugin Job Submit ?

Le plug-in CLI Filter s'exécute côté client sur les nœuds de connexion et de calcul avant que la soumission du travail n'atteigne le contrôleur, tandis que le plug-in Job Submit s'exécute côté serveur sur le contrôleur après la soumission du travail. Le plug-in CLI Filter peut être contourné par les utilisateurs mais ne verrouille pas le contrôleur, tandis que Job Submit est sécurisé mais peut avoir un impact sur les performances du cluster lors de son exécution.

Est-ce que AWS PCS supporte le plugin Slurm Job Submit ?

Non, le plugin Job Submit n'est pas pris en charge par AWS PCS. Utilisez plutôt le plugin CLI Filter pour la validation et la modification des soumissions de tâches.

Puis-je utiliser le plugin CLI Filter pour renforcer la sécurité ?

Non, le plugin CLI Filter peut être contourné par des utilisateurs déterminés et ne doit pas être utilisé pour renforcer la sécurité. Utilisez-le pour améliorer l'expérience utilisateur, définir les paramètres par défaut et fournir des conseils en matière de politiques plutôt que pour des politiques critiques en matière de sécurité.

Pourquoi le script doit-il se trouver sur tous les nœuds de calcul, et pas uniquement sur les nœuds de connexion ?

Des commandes comme Slurm `srun` peuvent être exécutées dans des scripts de travail sur des nœuds de calcul, ce qui déclenche également l'exécution du plugin CLI Filter. Le script doit être disponible partout où les commandes Slurm sont exécutées.

Puis-je modifier le script du plugin CLI Filter sur un cluster actif ?

Oui, si vous utilisez le S3 ou l'approche de déploiement du système de fichiers. Les nouvelles instances recevront le script mis à jour, mais les instances existantes doivent être mises à jour manuellement ou par le biais de la méthode de déploiement que vous avez choisie.

Puis-je utiliser différents scripts de plug-in CLI Filter sur différents groupes de nœuds de calcul ?

Oui, mais cela n'est pas recommandé. Vous pouvez fournir des scripts avec une logique différente aux différents groupes de nœuds de calcul, mais vous êtes responsable de la gestion des interdépendances et de la prévention des chevauchements logiques. La plupart des clients fournissent un ensemble de logique pour l'ensemble d'un cluster.

Puis-je utiliser le plugin CLI Filter avec une implémentation en C au lieu de Lua ?

L'implémentation en C n'est pas prise en charge. Seule l'implémentation de scripts Lua est prise en charge dans AWS PCS. SchedMD recommande aux clients d'utiliser Lua sur C pour faciliter l'utilisation lors de la mise en œuvre des plug-ins de filtre CLI.

Puis-je activer ou désactiver le plug-in CLI Filter sur un cluster existant ?

Oui, vous pouvez activer ou désactiver le plug-in CLI Filter sur les clusters existants à l'aide de l'API de mise à jour sans recréer le cluster.

Résolution des problèmes liés au plugin Slurm CLI Filter sur PCS AWS

Utilisez ces informations de dépannage pour résoudre les problèmes courants du plug-in CLI Filter.

La soumission du job échoue immédiatement avec une erreur de chargement du plugin

Symptômes : les utilisateurs reçoivent des messages d'erreur concernant l'absence ou l'échec du plug-in CLI Filter lorsqu'ils soumettent des tâches.

Causes possibles :

- Script du plugin CLI Filter absent d'un ou de plusieurs nœuds
- Nom de fichier de script incorrect (doit être exact`cli_filter.lua`)
- Script déployé sur le mauvais chemin de répertoire
- Le script possède des autorisations de fichier incorrectes

Résolution :

- Vérifiez que le script existe `/etc/aws/pcs/scheduler/slurm-<version>/cli_filter.lua` sur tous les nœuds de connexion et de calcul
- Vérifiez que le nom du fichier du script est exact `cli_filter.lua`
- Assurez-vous que le script possède des autorisations lisibles (644 ou similaires)
- Testez le déploiement du script sur un nœud de connexion unique avant de le déployer sur un cluster complet

La création du cluster échoue en raison d'une erreur de validation du plug-in CLI Filter

Symptômes : la création du cluster échoue en raison d'une erreur concernant un `CliFilterPlugins` paramètre non valide.

Causes possibles :

- Format de valeur de paramètre incorrect dans `slurmCustomSettings`
- Faute de frappe dans le nom ou la valeur du paramètre

Résolution :

- Utilisez le nom exact du paramètre : `CliFilterPlugins`
- Utilisez la valeur exacte du paramètre : `cli_filter/lua`
- Vérifier la syntaxe JSON dans le `slurmCustomSettings` tableau

Le script du plugin CLI Filter s'exécute mais la validation des tâches ne fonctionne pas comme prévu

Symptômes : les tâches sont soumises avec succès, mais la logique de validation personnalisée ne se déclenche pas ou ne produit pas de résultats inattendus.

Causes possibles :

- Erreurs de syntaxe du script Lua
- Modèles d'accès aux champs incorrects (utilisation de la syntaxe du plugin Job Submit au lieu du plug-in CLI Filter)
- Erreurs logiques dans les conditions de validation

Résolution :

- Vérifiez le script Lua pour détecter les erreurs de syntaxe
- Vérifiez que l'accès aux champs utilise le `options["field_name"]` format au lieu de `job_desc.field_name`
- Ajouter des instructions de journalisation au flux d'exécution du script de débogage
- Testez d'abord la logique du script avec des cas de validation simples

Le déploiement du script S3 échoue

Symptômes : les instances sont lancées mais le script du plug-in CLI Filter n'est pas téléchargé depuis S3.

Causes possibles :

- Le profil d'instance IAM ne dispose pas des autorisations de lecture S3
- Point de terminaison VPC S3 non configuré
- Chemin d'accès au compartiment ou à l'objet S3 incorrect dans les données utilisateur

Résolution :

- Vérifiez que le profil d'instance IAM est `s3:GetObject` autorisé à accéder à votre compartiment
- Configurer le point de terminaison S3 VPC Gateway pour un accès direct
- Vérifiez le nom du compartiment S3 et le chemin de l'objet dans le script de données utilisateur
- Vérifiez les journaux de données utilisateur de l'instance pour détecter les erreurs de téléchargement de S3

Sécurité dans le service de calcul AWS parallèle

La sécurité du cloud AWS est la priorité absolue. En tant que AWS client, vous bénéficiez de centres de données et d'architectures réseau conçus pour répondre aux exigences des entreprises les plus sensibles en matière de sécurité.

La sécurité est une responsabilité partagée entre vous AWS et vous. Le modèle de responsabilité partagée décrit ceci comme la sécurité du cloud et la sécurité dans le cloud :

- Sécurité du cloud : AWS est chargée de protéger l'infrastructure qui exécute les AWS services dans le AWS Cloud. AWS vous fournit également des services que vous pouvez utiliser en toute sécurité. Des auditeurs tiers testent et vérifient régulièrement l'efficacité de notre sécurité dans le cadre des programmes de AWS conformité Programmes de de conformité. Pour en savoir plus sur les programmes de conformité qui s'appliquent au service de calcul AWS parallèle, voir AWS Services concernés par programme de conformitéAWS .
- Sécurité dans le cloud — Votre responsabilité est déterminée par le AWS service que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris de la sensibilité de vos données, des exigences de votre entreprise, ainsi que de la législation et de la réglementation applicables.

Cette documentation vous aide à comprendre comment appliquer le modèle de responsabilité partagée lors de l'utilisation de AWS PCS. Les rubriques suivantes expliquent comment configurer les AWS PCS pour répondre à vos objectifs de sécurité et de conformité. Vous apprendrez également à utiliser d'autres AWS services qui vous aident à surveiller et à sécuriser les ressources de votre AWS PC.

Rubriques

- Protection des données dans le service de calcul AWS parallèle
- Accès AWS Parallel Computing Service via un point de terminaison d'interface (AWS PrivateLink)
- Identity and Access Management pour le service de calcul AWS parallèle
- Validation de conformité pour le service de calcul AWS parallèle
- Résilience dans les services de calcul AWS parallèle
- Sécurité de l'infrastructure dans un service de calcul AWS parallèle
- Analyse et gestion des vulnérabilités dans le service de calcul AWS parallèle
- Prévention du problème de l'adjoint confus entre services
- Bonnes pratiques de sécurité pour le service de calcul AWS parallèle

Protection des données dans le service de calcul AWS parallèle

Le modèle de responsabilité AWS partagée s'applique à la protection des données dans le service de calcul AWS parallèle. Comme décrit dans ce modèle, AWS est chargé de protéger l'infrastructure mondiale qui gère tous les AWS Cloud. La gestion du contrôle de votre contenu hébergé sur cette infrastructure relève de votre responsabilité. Vous êtes également responsable des tâches de configuration et de gestion de la sécurité des Services AWS que vous utilisez. Pour plus d'informations sur la confidentialité des données, consultez Questions fréquentes (FAQ) sur la confidentialité des données. Pour en savoir plus sur la protection des données en Europe, consultez le billet de blog Modèle de responsabilité partagée d'AWS et RGPD (Règlement général sur la protection des données) sur le Blog de sécuritéAWS .

À des fins de protection des données, nous vous recommandons de protéger les Compte AWS informations d'identification et de configurer les utilisateurs individuels avec AWS IAM Identity Center ou Gestion des identités et des accès AWS (IAM). Ainsi, chaque utilisateur se voit attribuer uniquement les autorisations nécessaires pour exécuter ses tâches. Nous vous recommandons également de sécuriser vos données comme indiqué ci-dessous :

- Utilisez l'authentification multifactorielle (MFA) avec chaque compte.
- SSL/TLS À utiliser pour communiquer avec AWS les ressources. Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Configurez l'API et la journalisation de l'activité des utilisateurs avec AWS CloudTrail. Pour plus d'informations sur l'utilisation des CloudTrail sentiers pour capturer AWS des activités, consultez la section Utilisation des CloudTrail sentiers dans le guide de AWS CloudTrail l'utilisateur.
- Utilisez des solutions de AWS chiffrement, ainsi que tous les contrôles de sécurité par défaut qu'ils contiennent Services AWS.
- Utilisez des services de sécurité gérés avancés tels qu'Amazon Macie, qui contribuent à la découverte et à la sécurisation des données sensibles stockées dans Amazon S3.
- Si vous avez besoin de modules cryptographiques validés par la norme FIPS 140-3 pour accéder AWS via une interface de ligne de commande ou une API, utilisez un point de terminaison FIPS. Pour plus d'informations sur les points de terminaison FIPS disponibles, consultez Norme FIPS (Federal Information Processing Standard) 140-3.

Nous vous recommandons fortement de ne jamais placer d'informations confidentielles ou sensibles, telles que les adresses e-mail de vos clients, dans des balises ou des champs de texte libre tels que le champ Nom. Cela inclut lorsque vous travaillez avec AWS PCS ou autre Services AWS à l'aide de

la console, de l'API ou AWS SDKs. AWS CLI Toutes les données que vous entrez dans des balises ou des champs de texte de forme libre utilisés pour les noms peuvent être utilisées à des fins de facturation ou dans les journaux de diagnostic. Si vous fournissez une adresse URL à un serveur externe, nous vous recommandons fortement de ne pas inclure d'informations d'identification dans l'adresse URL permettant de valider votre demande adressée à ce serveur.

Chiffrement au repos

Le chiffrement est activé par défaut pour les données au repos lorsque vous créez un cluster de services informatiques AWS parallèles (AWS PCS) avec l'API AWS Management Console AWS CLI,, AWS PCS ou AWS SDKs. AWS PCS utilise une clé KMS AWS détenue pour chiffrer les données au repos. Pour plus d'informations, consultez la section Clés client et AWS clés dans le guide du AWS KMS développeur. Vous pouvez également utiliser une clé gérée par le client. Pour de plus amples informations, veuillez consulter Politique de clé KMS requise pour une utilisation avec des volumes EBS chiffrés sur PCS AWS.

Le secret du cluster est stocké AWS Secrets Manager et chiffré avec la clé KMS gérée par Secrets Manager. Pour de plus amples informations, veuillez consulter Utilisation des secrets de cluster dans AWS PCS.

Dans un cluster AWS PCS, les données suivantes sont inactives :

- État du planificateur : il inclut des données sur les tâches en cours d'exécution et les nœuds provisionnés dans le cluster. Il s'agit des données que Slurm conserve telles que `StateSaveLocation` définies dans votre. `slurm.conf` Pour plus d'informations, consultez la description StateSaveLocationdans la documentation de Slurm. AWS PCS supprime les données des tâches une fois celles-ci terminées.
- Secret d'authentification du planificateur : AWS PCS l'utilise pour authentifier toutes les communications du planificateur dans le cluster.

Pour les informations sur l'état du planificateur, le AWS PCS chiffre automatiquement les données et les métadonnées avant de les écrire dans le système de fichiers. Le système de fichiers chiffré utilise l'algorithme de cryptage standard AES-256 pour les données au repos.

Chiffrement en transit

Vos connexions à l'API AWS PCS utilisent le cryptage TLS avec le processus de signature Signature Version 4, que vous utilisiez le AWS Command Line Interface (AWS CLI) ou AWS SDKs. Pour plus

d'informations, consultez la section Signature des demandes d' AWS API dans le guide de Gestion des identités et des accès AWS l'utilisateur. AWS gère le contrôle d'accès via l'API avec les politiques IAM relatives aux informations d'identification de sécurité que vous utilisez pour vous connecter.

AWS PCS utilise le protocole TLS pour se connecter à d'autres AWS services.

Au sein d'un cluster Slurm, le planificateur est configuré avec le plug-in `auth/slurm` d'authentification qui fournit une authentification pour toutes les communications du planificateur. Slurm ne fournit pas de chiffrement au niveau de l'application pour ses communications, toutes les données circulant entre les instances de cluster restent locales dans le VPC EC2 et sont donc soumises au chiffrement VPC si ces instances prennent en charge le chiffrement en transit. Pour plus d'informations, consultez la section Chiffrement en transit dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud. La communication est cryptée entre le contrôleur (fourni dans un compte de service) et les nœuds du cluster de votre compte.

Gestion des clés

AWS PCS utilise une clé KMS AWS détenue pour chiffrer les données. Pour plus d'informations, consultez la section Clés client et AWS clés dans le guide du AWS KMS développeur. Vous pouvez également utiliser une clé gérée par le client. Pour de plus amples informations, veuillez consulter Politique de clé KMS requise pour une utilisation avec des volumes EBS chiffrés sur PCS AWS.

Le secret du cluster est stocké AWS Secrets Manager et chiffré avec la clé KMS gérée par Secrets Manager. Pour de plus amples informations, veuillez consulter Utilisation des secrets de cluster dans AWS PCS.

Confidentialité du trafic inter-réseaux

AWS Les ressources de calcul PCS d'un cluster se trouvent dans 1 VPC sur le compte du client. Par conséquent, tout le trafic interne du service AWS PCS au sein d'un cluster reste sur le AWS réseau et ne circule pas sur Internet. La communication entre l'utilisateur et les nœuds AWS PCS peut se faire via Internet et nous vous recommandons d'utiliser SSH ou Systems Manager pour vous connecter aux nœuds. Pour plus d'informations, voir Qu'est-ce que c'est AWS Systems Manager ? dans le guide de AWS Systems Manager l'utilisateur.

Vous pouvez également utiliser les offres suivantes pour connecter votre réseau local à AWS :

- AWS Site-to-Site VPN. Pour plus d'informations, voir Qu'est-ce que c'est AWS Site-to-Site VPN ? dans le guide de AWS Site-to-Site VPN l'utilisateur.

- Un AWS Direct Connect. Pour plus d'informations, voir Qu'est-ce que c'est AWS Direct Connect ? dans le guide de AWS Direct Connect l'utilisateur.

Vous accédez à l'API AWS PCS pour effectuer des tâches administratives pour le service. Vous et vos utilisateurs accédez aux ports du point de terminaison Slurm pour interagir directement avec le planificateur.

Chiffrer le trafic des API

Pour accéder à l'API AWS PCS, les clients doivent prendre en charge le protocole TLS (Transport Layer Security) 1.2 ou version ultérieure. Nous exigeons TLS 1.2 et recommandons TLS 1.3. Les clients doivent également prendre en charge les suites de chiffrement PFS (Perfect Forward Secrecy) comme Ephemeral Diffie-Hellman (DHE) ou Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes. En outre, les demandes doivent être signées à l'aide d'un ID de clé d'accès et d'une clé d'accès secrète associée à un principal IAM. Vous pouvez également utiliser AWS Security Token Service (AWS STS) pour générer des informations de sécurité temporaires afin de signer les demandes.

Chiffrement du trafic de données

Le chiffrement des données en transit est activé à partir des instances EC2 prises en charge accédant au point de terminaison du planificateur et entre les ComputeNodeGroup instances depuis le. AWS Cloud Pour de plus amples informations, veuillez consulter Chiffrement en transit.

Politique de clé KMS requise pour une utilisation avec des volumes EBS chiffrés sur PCS AWS

AWS PCS utilise des rôles liés aux services pour déléguer des autorisations à d'autres. Services AWS Le rôle lié au service AWS PCS est prédéfini et inclut les autorisations dont AWS PCS a besoin pour appeler d'autres personnes en votre Services AWS nom. Les autorisations prédéfinies incluent également l'accès à vos clés gérées par le client Clés gérées par AWS , mais pas à celles gérées par vos clients.

Cette rubrique explique comment configurer la politique de clé requise pour lancer des instances lorsque vous spécifiez une clé gérée par le client pour le chiffrement Amazon EBS.

> **Note**
>
> AWS PCS n'a pas besoin d'autorisation supplémentaire pour utiliser la valeur par défaut Clé gérée par AWS afin de protéger les volumes chiffrés de votre compte.

Table des matières

- Présentation de
- Configurer des politiques de clé
- Exemple 1 : sections de la politique de clé qui autorisent l'accès à la clé gérée par le client
- Exemple 2 : sections de la politique de clé autorisant l'accès entre comptes à la clé gérée par le client
- Modifier les politiques clés dans la AWS KMS console

Présentation de

Vous pouvez utiliser les éléments suivants AWS KMS keys pour le chiffrement Amazon EBS lorsque AWS PCS lance des instances :

- Clé gérée par AWS— Une clé de chiffrement dans votre compte créée, détenue et gérée par Amazon EBS. Il s'agit de la clé de chiffrement par défaut d'un nouveau compte. Amazon EBS utilise le Clé gérée par AWS pour le chiffrement, sauf si vous spécifiez une clé gérée par le client.
- Clé gérée par le client : clé de chiffrement personnalisée que vous créez, détenez et gérez vous-même. Pour plus d'informations, consultez la section Création d'une clé KMS dans le guide du AWS Key Management Service développeur.

> **Note**
>
> La clé doit être symétrique. Amazon EBS ne prend pas en charge les clés asymétriques gérées par le client.

Vous configurez les clés gérées par le client lorsque vous créez des instantanés chiffrés ou un modèle de lancement qui spécifie des volumes chiffrés, ou lorsque vous choisissez d'activer le chiffrement par défaut.

Configurer des politiques de clé

Vos clés KMS doivent avoir une politique clé qui permet à AWS PCS de lancer des instances avec des volumes Amazon EBS chiffrés à l'aide d'une clé gérée par le client.

Utilisez les exemples de cette page pour configurer une politique de clé afin de permettre à AWS PCS d'accéder à votre clé gérée par le client. Vous pouvez modifier la politique clé de la clé gérée par le client lors de la création de la clé ou ultérieurement.

La politique clé doit comporter les énoncés suivants :

- Une instruction qui permet à l'identité IAM spécifiée dans l'`Principal`élément d'utiliser directement la clé gérée par le client. Il inclut les autorisations permettant d'effectuer les `DescribeKey` opérations AWS KMS `Encrypt` `DecryptReEncrypt*`,`GenerateDataKey*`,, et sur la clé.
- Une instruction qui permet à l'identité IAM spécifiée dans l'`Principal`élément d'utiliser l'`CreateGrant`opération pour générer des autorisations déléguant un sous-ensemble de ses propres autorisations à des entités intégrées à Services AWS un autre principal AWS KMS ou à un autre. Il est ainsi possible d'utiliser la clé pour créer des ressources chiffrées en votre nom.

Ne modifiez aucune déclaration existante dans la politique lorsque vous ajoutez les nouvelles déclarations de politique à votre politique principale.

Pour en savoir plus, consultez :

- create-key dans la référence de commande AWS CLI
- put-key-policy dans la référence de commande de l'AWS CLI
- Trouvez l'ID et l'ARN de la clé dans le guide du AWS Key Management Service développeur
- Rôles liés aux services pour PCS AWS
- Chiffrement Amazon EBS dans le guide de l'utilisateur Amazon EBS
- AWS Key Management Servicedans le guide AWS Key Management Service du développeur

Exemple 1 : sections de la politique de clé qui autorisent l'accès à la clé gérée par le client

Ajoutez les déclarations de politique suivantes à la politique clé de la clé gérée par le client. Remplacez l'exemple d'ARN par l'ARN du rôle `AWSServiceRoleForPCS` lié à votre service.

Cet exemple de politique donne au rôle lié au service AWS PCS (`AWSServiceRoleForPCS`) l'autorisation d'utiliser la clé gérée par le client.

```
{
   "Sid": "Allow service-linked role use of the customer managed key",
   "Effect": "Allow",
   "Principal": {
       "AWS": [
           "arn:aws:iam::account-id:role/aws-service-role/pcs.amazonaws.com/
AWSServiceRoleForPCS"
       ]
   },
   "Action": [
       "kms:Encrypt",
       "kms:Decrypt",
       "kms:ReEncrypt*",
       "kms:GenerateDataKey*",
       "kms:DescribeKey"
   ],
   "Resource": "*"
}
```

```
{
   "Sid": "Allow attachment of persistent resources",
   "Effect": "Allow",
   "Principal": {
       "AWS": [
           "arn:aws:iam::account-id:role/aws-service-role/pcs.amazonaws.com/
AWSServiceRoleForPCS"
       ]
   },
   "Action": [
       "kms:CreateGrant"
   ],
   "Resource": "*",
   "Condition": {
       "Bool": {
           "kms:GrantIsForAWSResource": true
       }
    }
}
```

Exemple 2 : sections de la politique de clé autorisant l'accès entre comptes à la clé gérée par le client

Si vous créez une clé gérée par le client dans un compte différent de celui de votre cluster AWS PCS, vous devez utiliser une autorisation en combinaison avec la politique des clés pour autoriser l'accès entre comptes à la clé.

Pour autoriser l'accès à la clé

1. Ajoutez les déclarations de politique suivantes à la politique clé de la clé gérée par le client. Remplacez l'exemple d'ARN par l'ARN de l'autre compte. *111122223333*Remplacez-le par l'ID de compte réel du compte dans Compte AWS lequel vous souhaitez créer le cluster AWS PCS. Cela vous permet de donner à un utilisateur ou à un rôle IAM du compte spécifié l'autorisation de créer un octroi pour la clé à l'aide de la commande CLI suivante. Par défaut, les utilisateurs n'ont pas accès à la clé.

```
{.
   "Sid": "Allow external account 111122223333 use of the customer managed key",
   "Effect": "Allow",
   "Principal": {
       "AWS": [
           "arn:aws:iam::111122223333:root"
       ]
   },
   "Action": [
       "kms:Encrypt",
       "kms:Decrypt",
       "kms:ReEncrypt*",
       "kms:GenerateDataKey*",
       "kms:DescribeKey"
   ],
   "Resource": "*"
}
```

```
{
   "Sid": "Allow attachment of persistent resources in external
 account 111122223333",
   "Effect": "Allow",
   "Principal": {
       "AWS": [
           "arn:aws:iam::111122223333:root"
```

```
        ]
    },
    "Action": [
        "kms:CreateGrant"
    ],
    "Resource": "*"
}
```

2. À partir du compte dans lequel vous souhaitez créer le cluster AWS PCS, créez une subvention qui délègue les autorisations pertinentes au rôle lié au service AWS PCS. La valeur de `grantee-principal` est l'ARN du rôle lié au service. La valeur de `key-id` est l'ARN de la clé.

   L'exemple de commande CLI create-grant suivant donne au rôle lié au service nommé `AWSServiceRoleForPCS` dans le compte l'*111122223333*autorisation d'utiliser la clé gérée par le client dans le compte. *444455556666*

```
aws kms create-grant \
  --region us-west-2 \
  --key-id arn:aws:kms:us-
west-2:444455556666:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d \
  --grantee-principal arn:aws:iam::111122223333:role/aws-service-role/
pcs.amazonaws.com/AWSServiceRoleForPCS \
  --operations "Encrypt" "Decrypt" "ReEncryptFrom" "ReEncryptTo" "GenerateDataKey"
 "GenerateDataKeyWithoutPlaintext" "DescribeKey" "CreateGrant"
```

   > **Note**
   >
   > L'utilisateur qui fait la demande doit être autorisé à utiliser l'`kms:CreateGrant`action.

   L'exemple de politique IAM suivant permet à une identité IAM (utilisateur ou rôle) dans le compte *111122223333* de créer une autorisation pour la clé gérée par le client dans le compte. *444455556666*

   JSON

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

```
      "Sid": "AllowCreationOfGrantForTheKMSKeyinExternalAccount444455556666",
      "Effect": "Allow",
      "Action": "kms:CreateGrant",
      "Resource": "arn:aws:kms:us-
west-2:444455556666:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    }
  ]
}
```

Pour plus d'informations sur la création d'un octroi pour une clé KMS dans un autre Compte AWS, consultez la rubrique Octrois dans AWS KMS dans le Guide du développeur AWS Key Management Service .

> **Important**
>
> Le nom du rôle lié au service spécifié en tant que principal bénéficiaire doit être le nom d'un rôle existant. Après avoir créé la subvention, pour vous assurer qu'elle autorise AWS PCS à utiliser la clé KMS spécifiée, ne supprimez pas et ne recréez pas le rôle lié au service.

Modifier les politiques clés dans la AWS KMS console

Les exemples des sections précédentes montrent comment ajouter des déclarations à une politique de clé, qui est simplement un moyen de modifier une politique de clé. Le moyen le plus simple de modifier une politique clé consiste à utiliser la vue par défaut de la AWS KMS console pour les politiques clés et à faire d'une identité IAM (utilisateur ou rôle) l'un des principaux utilisateurs de la stratégie clé appropriée. Pour plus d'informations, consultez la section Utilisation de l'affichage AWS Management Console par défaut dans le Guide du AWS Key Management Service développeur.

> **Warning**
>
> Les déclarations de politique d'affichage par défaut de la console incluent les autorisations permettant d'effectuer AWS KMS `Revoke` des opérations sur la clé gérée par le client. Si vous révoquez une autorisation qui donnait Compte AWS accès à une clé gérée par le client dans votre compte, les utilisateurs de ce compte Compte AWS perdent l'accès aux données cryptées et à la clé.

Accès AWS Parallel Computing Service via un point de terminaison d'interface (AWS PrivateLink)

Vous pouvez l'utiliser AWS PrivateLink pour créer une connexion privée entre votre VPC et AWS Parallel Computing Service ()AWS PCS. Vous pouvez y accéder AWS PCS comme s'il se trouvait dans votre VPC, sans utiliser de passerelle Internet, de périphérique NAT, de connexion VPN ou Direct Connect de connexion. Les instances de votre VPC n'ont pas besoin d'adresses IP publiques pour y accéder. AWS PCS

Vous établissez cette connexion privée en créant un point de terminaison d'interface optimisé par AWS PrivateLink. Nous créons une interface réseau de point de terminaison dans chaque sous-réseau que vous activez pour le point de terminaison d'interface. Il s'agit d'interfaces réseau gérées par le demandeur qui servent de point d'entrée pour le trafic destiné à AWS PCS.

Pour plus d'informations, consultez la section Accès Services AWS par AWS PrivateLink le biais du AWS PrivateLink guide.

Considérations relatives à AWS PCS

Avant de configurer un point de terminaison d'interface pour AWS PCS, consultez la section Accès à un service AWS à l'aide d'un point de terminaison VPC d'interface dans le AWS PrivateLink Guide.

AWS PCS prend en charge les appels à toutes ses actions d'API via le point de terminaison de l'interface.

Si votre VPC ne dispose pas d'un accès direct à Internet, vous devez configurer un point de terminaison VPC pour permettre aux instances de votre groupe de nœuds de calcul d'appeler l'action d'API. AWS PCS `RegisterComputeNodeGroupInstance`

Créez un point de terminaison d'interface pour AWS PCS

Vous pouvez créer un point de terminaison d'interface pour AWS PCS utiliser la console Amazon VPC ou le AWS Command Line Interface ()AWS CLI. Pour plus d'informations, consultez Création d'un point de terminaison d'interface dans le Guide AWS PrivateLink .

Créez un point de terminaison d'interface pour AWS PCS utiliser le nom de service suivant :

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

Remplacez *region* par l'ID du dans Région AWS lequel créer le point de terminaison, tel que `us-east-1`.

Si vous activez le DNS privé pour le point de terminaison de l'interface, vous pouvez envoyer des demandes d'API à AWS PCS l'aide de son nom DNS régional par défaut. Par exemple, `pcs.us-east-1.amazonaws.com`.

Création d'une politique de point de terminaison pour votre point de terminaison d'interface

Une politique de point de terminaison est une ressource IAM que vous pouvez attacher à votre point de terminaison d'interface. La politique de point de terminaison par défaut autorise un accès complet AWS PCS via le point de terminaison de l'interface. Pour contrôler l'accès autorisé AWS PCS depuis votre VPC, associez une politique de point de terminaison personnalisée au point de terminaison de l'interface.

Une politique de point de terminaison spécifie les informations suivantes :

- Les principaux qui peuvent effectuer des actions (Comptes AWS, utilisateurs IAM et rôles IAM).
- Les actions qui peuvent être effectuées.
- La ressource sur laquelle les actions peuvent être effectuées.

Pour plus d'informations, consultez Contrôle de l'accès aux services à l'aide de politiques de point de terminaison dans le Guide AWS PrivateLink .

Exemple : politique de point de terminaison VPC pour les actions AWS PCS

Voici un exemple de politique de point de terminaison personnalisée. Lorsque vous attachez cette politique au point de terminaison de votre interface, elle accorde l'accès aux AWS PCS actions répertoriées pour tous les principaux du cluster ayant la valeur spécifiée*cluster-id*. Remplacez *region* par l'ID Région AWS du cluster, tel que`us-east-1`. Remplacez *account-id* par le Compte AWS numéro du 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 pour le service de calcul AWS parallèle

Gestion des identités et des accès AWS (IAM) est un outil Service AWS qui permet à un administrateur de contrôler en toute sécurité l'accès aux AWS ressources. Les administrateurs IAM contrôlent qui peut être authentifié (connecté) et autorisé (autorisé) à utiliser les ressources AWS PCS. IAM est un Service AWS outil que vous pouvez utiliser sans frais supplémentaires.

Rubriques

- Public ciblé
- Authentification par des identités
- Gestion de l'accès à l'aide de politiques
- Comment fonctionne AWS Parallel Computing Service avec IAM
- Exemples de politiques basées sur l'identité pour le service de calcul AWS parallèle
- AWS politiques gérées pour le service de calcul AWS parallèle
- Rôles liés aux services pour PCS AWS
- Rôle Amazon EC2 Spot pour PC AWS
- Autorisations minimales pour les AWS PCS
- Profils d'instance IAM pour AWS Parallel Computing Service
- Résolution des problèmes d'identité et d'accès au service de calcul AWS parallèle

Public ciblé

La façon dont vous utilisez Gestion des identités et des accès AWS (IAM) varie en fonction de votre rôle :

- Utilisateur du service : demandez des autorisations à votre administrateur si vous ne pouvez pas accéder aux fonctionnalités (voir Résolution des problèmes d'identité et d'accès au service de calcul AWS parallèle)
- Administrateur du service : déterminez l'accès des utilisateurs et soumettez les demandes d'autorisation (voir Comment fonctionne AWS Parallel Computing Service avec IAM)
- Administrateur IAM : rédigez des politiques pour gérer l'accès (voir Exemples de politiques basées sur l'identité pour le service de calcul AWS parallèle)

Authentification par des identités

L'authentification est la façon dont vous vous connectez à AWS l'aide de vos informations d'identification. Vous devez être authentifié en tant qu'utilisateur IAM ou en assumant un rôle IAM. Utilisateur racine d'un compte AWS

Vous pouvez vous connecter en tant qu'identité fédérée à l'aide d'informations d'identification provenant d'une source d'identité telle que AWS IAM Identity Center (IAM Identity Center), d'une authentification unique ou d'informations d'identification. Google/Facebook Pour plus d'informations sur la connexion, consultez Connexion à votre Compte AWS dans le Guide de l'utilisateur Connexion à AWS .

Pour l'accès par programmation, AWS fournit un SDK et une CLI pour signer les demandes de manière cryptographique. Pour plus d'informations, consultez Signature AWS Version 4 pour les demandes d'API dans le Guide de l'utilisateur IAM.

Compte AWS utilisateur root

Lorsque vous créez un Compte AWS, vous commencez par une seule identité de connexion appelée utilisateur Compte AWS root qui dispose d'un accès complet à toutes Services AWS les ressources. Il est vivement déconseillé d'utiliser l'utilisateur racine pour vos tâches quotidiennes. Pour les tâches qui requièrent des informations d'identification de l'utilisateur racine, consultez Tâches qui requièrent les informations d'identification de l'utilisateur racine dans le Guide de l'utilisateur IAM.

Identité fédérée

Il est recommandé d'obliger les utilisateurs humains à utiliser la fédération avec un fournisseur d'identité pour accéder à Services AWS l'aide d'informations d'identification temporaires.

Une identité fédérée est un utilisateur provenant de l'annuaire de votre entreprise, de votre fournisseur d'identité Web ou Directory Service qui y accède à Services AWS l'aide d'informations d'identification provenant d'une source d'identité. Les identités fédérées assument des rôles qui fournissent des informations d'identification temporaires.

Pour une gestion des accès centralisée, nous vous recommandons d'utiliser AWS IAM Identity Center. Pour plus d'informations, consultez Qu'est-ce que IAM Identity Center ? dans le Guide de l'utilisateur AWS IAM Identity Center .

Utilisateurs et groupes IAM

Un utilisateur IAM est une identité qui dispose d'autorisations spécifiques pour une seule personne ou application. Nous vous recommandons d'utiliser ces informations d'identification temporaires au lieu des utilisateurs IAM avec des informations d'identification à long terme. Pour plus d'informations, voir Exiger des utilisateurs humains qu'ils utilisent la fédération avec un fournisseur d'identité pour accéder à AWS l'aide d'informations d'identification temporaires dans le guide de l'utilisateur IAM.

Les groupes IAM spécifient une collection d'utilisateurs IAM et permettent de gérer plus facilement les autorisations pour de grands ensembles d'utilisateurs. Pour plus d'informations, consultez Cas d'utilisation pour les utilisateurs IAM dans le Guide de l'utilisateur IAM.

Rôles IAM

Un rôle IAM est une identité dotée d'autorisations spécifiques qui fournit des informations d'identification temporaires. Vous pouvez assumer un rôle en passant d'un rôle utilisateur à un rôle IAM (console) ou en appelant une opération AWS CLI ou AWS API. Pour plus d'informations, consultez Méthodes pour endosser un rôle dans le Guide de l'utilisateur IAM.

Les rôles IAM sont utiles pour l'accès des utilisateurs fédérés, les autorisations temporaires des utilisateurs IAM, les accès intercompte, les accès entre services et les applications exécutées sur Amazon EC2. Pour plus d'informations, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Gestion de l’accès à l’aide de politiques

Vous contrôlez l'accès en AWS créant des politiques et en les associant à AWS des identités ou à des ressources. Une politique définit les autorisations lorsqu'elles sont associées à une identité ou à une ressource. AWS évalue ces politiques lorsqu'un directeur fait une demande. La plupart des politiques sont stockées AWS sous forme de documents JSON. Pour plus d’informations les documents de politique JSON, consultez Vue d’ensemble des politiques JSON dans le Guide de l’utilisateur IAM.

À l’aide de politiques, les administrateurs précisent qui a accès à quoi en définissant quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

Par défaut, les utilisateurs et les rôles ne disposent d’aucune autorisation. Un administrateur IAM crée des politiques IAM et les ajoute aux rôles, que les utilisateurs peuvent ensuite assumer. Les politiques IAM définissent les autorisations quelle que soit la méthode que vous utilisez pour exécuter l’opération.

Politiques basées sur l’identité

Les stratégies basées sur l’identité sont des documents de stratégie d’autorisations JSON que vous attachez à une identité (utilisateur, groupe ou rôle). Ces politiques contrôlent les actions que peuvent exécuter ces identités, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l’identité, consultez Définition d’autorisations IAM personnalisées avec des politiques gérées par le client dans le Guide de l’utilisateur IAM.

Les politiques basées sur l’identité peuvent être des politiques intégrées (intégrées directement dans une seule identité) ou des politiques gérées (politiques autonomes associées à plusieurs identités). Pour découvrir comment choisir entre des politiques gérées et en ligne, consultez Choix entre les politiques gérées et les politiques en ligne dans le Guide de l’utilisateur IAM.

Politiques basées sur les ressources

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Les exemples incluent les politiques de confiance de rôle IAM et les stratégies de compartiment Amazon S3. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l’accès à une ressource spécifique. Vous devez spécifier un principal dans une politique basée sur les ressources.

Les politiques basées sur les ressources sont des politiques en ligne situées dans ce service. Vous ne pouvez pas utiliser les politiques AWS gérées par IAM dans une stratégie basée sur les ressources.

Autres types de politique

AWS prend en charge des types de politiques supplémentaires qui peuvent définir les autorisations maximales accordées par les types de politiques les plus courants :

- Limites d'autorisations : une limite des autorisations définit le nombre maximum d'autorisations qu'une politique basée sur l'identité peut accorder à une entité IAM. Pour plus d'informations, consultez Limites d'autorisations pour des entités IAM dans le Guide de l'utilisateur IAM.
- Politiques de contrôle des services (SCPs) — Spécifiez les autorisations maximales pour une organisation ou une unité organisationnelle dans AWS Organizations. Pour plus d'informations, consultez Politiques de contrôle de service dans le Guide de l'utilisateur AWS Organizations .
- Politiques de contrôle des ressources (RCPs) : définissez le maximum d'autorisations disponibles pour les ressources de vos comptes. Pour plus d'informations, voir Politiques de contrôle des ressources (RCPs) dans le guide de AWS Organizations l'utilisateur.
- Politiques de session : politiques avancées que vous passez en tant que paramètre lorsque vous créez par programmation une session temporaire pour un rôle ou un utilisateur fédéré. Pour plus d'informations, consultez Politiques de session dans le Guide de l'utilisateur IAM.

Plusieurs types de politique

Lorsque plusieurs types de politiques s'appliquent à la requête, les autorisations en résultant sont plus compliquées à comprendre. Pour savoir comment AWS déterminer s'il faut autoriser une demande lorsque plusieurs types de politiques sont impliqués, consultez la section Logique d'évaluation des politiques dans le guide de l'utilisateur IAM.

Comment fonctionne AWS Parallel Computing Service avec IAM

Avant d'utiliser IAM pour gérer l'accès aux AWS PCS, découvrez quelles fonctionnalités IAM peuvent être utilisées avec AWS les PCS.

Fonctionnalités IAM que vous pouvez utiliser avec AWS Parallel Computing Service

Fonctionnalité IAM	AWS Support PCS
Politiques basées sur l'identité	Oui
Politiques basées sur les ressources	Non
Actions de politique	Oui
Ressources de politique	Oui
Clés de condition de politique (spécifiques au service)	Oui
ACLs	Non
ABAC (étiquettes dans les politiques)	Oui
Informations d'identification temporaires	Oui
Autorisations de principal	Oui
Rôles du service	Non
Rôles liés à un service	Oui

Pour obtenir une vue d'ensemble de la façon dont les AWS PCS et les autres AWS services fonctionnent avec la plupart des fonctionnalités IAM, consultez la section AWS Services compatibles avec IAM dans le Guide de l'utilisateur IAM.

Politiques basées sur l'identité pour les PCS AWS

Prend en charge les politiques basées sur l'identité : oui

Les politiques basées sur l'identité sont des documents de politique d'autorisations JSON que vous pouvez attacher à une identité telle qu'un utilisateur, un groupe d'utilisateurs ou un rôle IAM. Ces politiques contrôlent quel type d'actions des utilisateurs et des rôles peuvent exécuter, sur quelles ressources et dans quelles conditions. Pour découvrir comment créer une politique basée sur l'identité, consultez Définition d'autorisations IAM personnalisées avec des politiques gérées par le client dans le Guide de l'utilisateur IAM.

Avec les politiques IAM basées sur l'identité, vous pouvez spécifier des actions et ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. Pour découvrir tous les éléments que vous utilisez dans une politique JSON, consultez Références des éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.

Exemples de politiques basées sur l'identité pour PCS AWS

Pour consulter des exemples de politiques basées sur l'identité AWS PCS, consultez. Exemples de politiques basées sur l'identité pour le service de calcul AWS parallèle

Politiques basées sur les ressources au sein du PCS AWS

Prend en charge les politiques basées sur les ressources : non

Les politiques basées sur les ressources sont des documents de politique JSON que vous attachez à une ressource. Par exemple, les politiques de confiance de rôle IAM et les politiques de compartiment Amazon S3 sont des politiques basées sur les ressources. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l'accès à une ressource spécifique. Pour la ressource dans laquelle se trouve la politique, cette dernière définit quel type d'actions un principal spécifié peut effectuer sur cette ressource et dans quelles conditions. Vous devez spécifier un principal dans une politique basée sur les ressources. Les principaux peuvent inclure des comptes, des utilisateurs, des rôles, des utilisateurs fédérés ou. Services AWS

Pour permettre un accès intercompte, vous pouvez spécifier un compte entier ou des entités IAM dans un autre compte en tant que principal dans une politique basée sur les ressources. Pour plus d'informations, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Actions politiques pour le AWS PCS

Prend en charge les actions de politique : oui

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Action` d'une politique JSON décrit les actions que vous pouvez utiliser pour autoriser ou refuser l'accès à une politique. Intégration d'actions dans une politique afin d'accorder l'autorisation d'exécuter les opérations associées.

Pour consulter la liste des actions AWS PCS, voir Actions définies par le service de calcul AWS parallèle dans la référence d'autorisation de service.

Les actions de politique dans AWS PCS utilisent le préfixe suivant avant l'action :

```
pcs
```

Pour indiquer plusieurs actions dans une seule déclaration, séparez-les par des virgules.

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

Ressources relatives aux politiques pour le AWS PCS

Prend en charge les ressources de politique : oui

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément de politique JSON `Resource` indique le ou les objets auxquels l'action s'applique. Il est recommandé de définir une ressource à l'aide de son Amazon Resource Name (ARN). Pour les actions qui ne sont pas compatibles avec les autorisations de niveau ressource, utilisez un caractère générique (*) afin d'indiquer que l'instruction s'applique à toutes les ressources.

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

Pour consulter la liste des types de ressources AWS PCS et leurs caractéristiques ARNs, consultez la section Ressources définies par le service de calcul AWS parallèle dans la référence d'autorisation du service. Pour savoir avec quelles actions vous pouvez spécifier l'ARN de chaque ressource, voir Actions définies par le service de calcul AWS parallèle.

Pour consulter des exemples de politiques basées sur l'identité AWS PCS, consultez. Exemples de politiques basées sur l'identité pour le service de calcul AWS parallèle

Clés d'état des politiques pour AWS PCS

Prend en charge les clés de condition de politique spécifiques au service : oui

Les administrateurs peuvent utiliser les politiques AWS JSON pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Condition` indique à quel moment les instructions s'exécutent en fonction de critères définis. Vous pouvez créer des expressions conditionnelles qui utilisent des opérateurs de condition, tels que les signes égal ou inférieur à, pour faire correspondre la condition de la politique aux valeurs de la demande. Pour voir toutes les clés de condition AWS globales, voir les clés de contexte de condition AWS globales dans le guide de l'utilisateur IAM.

Pour consulter la liste des clés de condition AWS PCS, voir Clés de condition pour le service de calcul AWS parallèle dans la référence d'autorisation de service. Pour savoir avec quelles actions et ressources vous pouvez utiliser une clé de condition, voir Actions définies par AWS Parallel Computing Service.

Pour consulter des exemples de politiques basées sur l'identité AWS PCS, consultez. Exemples de politiques basées sur l'identité pour le service de calcul AWS parallèle

ACLs en AWS PCS

Supports ACLs : Non

Les listes de contrôle d'accès (ACLs) contrôlent les principaux (membres du compte, utilisateurs ou rôles) autorisés à accéder à une ressource. ACLs sont similaires aux politiques basées sur les ressources, bien qu'elles n'utilisent pas le format de document de politique JSON.

ABAC avec PCS AWS

Prise en charge d'ABAC (balises dans les politiques) : Oui

Le contrôle d'accès par attributs (ABAC) est une stratégie d'autorisation qui définit les autorisations en fonction des attributs appelés balises. Vous pouvez associer des balises aux entités et aux AWS ressources IAM, puis concevoir des politiques ABAC pour autoriser les opérations lorsque la balise du principal correspond à la balise de la ressource.

Pour contrôler l'accès basé sur des étiquettes, vous devez fournir les informations d'étiquette dans l'élément de condition d'une politique utilisant les clés de condition `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* ou `aws:TagKeys`.

Si un service prend en charge les trois clés de condition pour tous les types de ressources, alors la valeur pour ce service est Oui. Si un service prend en charge les trois clés de condition pour certains types de ressources uniquement, la valeur est Partielle.

Pour plus d'informations sur ABAC, consultez Définition d'autorisations avec l'autorisation ABAC dans le Guide de l'utilisateur IAM. Pour accéder à un didacticiel décrivant les étapes de configuration de l'ABAC, consultez Utilisation du contrôle d'accès par attributs (ABAC) dans le Guide de l'utilisateur IAM.

Utilisation d'informations d'identification temporaires avec AWS PCS

Prend en charge les informations d'identification temporaires : oui

Les informations d'identification temporaires fournissent un accès à court terme aux AWS ressources et sont automatiquement créées lorsque vous utilisez la fédération ou que vous changez de rôle. AWS recommande de générer dynamiquement des informations d'identification temporaires au lieu d'utiliser des clés d'accès à long terme. Pour plus d'informations, consultez Informations d'identification de sécurité temporaires dans IAM et Services AWS compatibles avec IAM dans le Guide de l'utilisateur IAM.

Autorisations principales interservices pour PCS AWS

Prend en charge les sessions d'accès direct (FAS) : oui

Les sessions d'accès direct (FAS) utilisent les autorisations du principal appelant et Service AWS, combinées Service AWS à la demande d'envoi de demandes aux services en aval. Pour plus de détails sur la politique relative à la transmission de demandes FAS, consultez la section Sessions de transmission d'accès.

Rôles de service pour AWS PCS

Prend en charge les rôles de service : Non

Un rôle de service est un rôle IAM qu'un service endosse pour accomplir des actions en votre nom. Un administrateur IAM peut créer, modifier et supprimer un rôle de service à partir d'IAM. Pour plus d'informations, consultez Création d'un rôle pour la délégation d'autorisations à un Service AWS dans le Guide de l'utilisateur IAM.

> **Warning**
>
> La modification des autorisations associées à un rôle de service peut perturber les fonctionnalités du AWS PCS. Modifiez les rôles de service uniquement lorsque AWS PCS fournit des instructions à cet effet.

Rôles liés aux services pour PCS AWS

Prend en charge les rôles liés à un service : oui

Un rôle lié à un service est un type de rôle de service lié à un. Service AWS Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés à un service apparaissent dans votre Compte AWS répertoire et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier, les autorisations concernant les rôles liés à un service.

Pour plus de détails sur la création ou la gestion des rôles liés aux services AWS PCS, consultez. Rôles liés aux services pour PCS AWS

Exemples de politiques basées sur l'identité pour le service de calcul AWS parallèle

Par défaut, les utilisateurs et les rôles ne sont pas autorisés à créer ou à modifier des ressources AWS PCS. Pour octroyer aux utilisateurs des autorisations d'effectuer des actions sur les ressources dont ils ont besoin, un administrateur IAM peut créer des politiques IAM.

Pour apprendre à créer une politique basée sur l'identité IAM à l'aide de ces exemples de documents de politique JSON, consultez Création de politiques IAM (console) dans le Guide de l'utilisateur IAM.

Pour plus de détails sur les actions et les types de ressources définis par AWS PCS, y compris le ARNs format de chaque type de ressource, voir Actions, ressources et clés de condition pour le service de calcul AWS parallèle dans la référence d'autorisation du service.

Rubriques

- Bonnes pratiques en matière de politiques
- Utilisation de la console AWS PCS
- Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Bonnes pratiques en matière de politiques

Les politiques basées sur l'identité déterminent si quelqu'un peut créer, accéder ou supprimer des ressources AWS PCS dans votre compte. Ces actions peuvent entraîner des frais pour votre Compte AWS. Lorsque vous créez ou modifiez des politiques basées sur l'identité, suivez ces instructions et recommandations :

- Commencez AWS par les politiques gérées et passez aux autorisations du moindre privilège : pour commencer à accorder des autorisations à vos utilisateurs et à vos charges de travail, utilisez les politiques AWS gérées qui accordent des autorisations pour de nombreux cas d'utilisation courants. Ils sont disponibles dans votre Compte AWS. Nous vous recommandons de réduire davantage les autorisations en définissant des politiques gérées par les AWS clients spécifiques à vos cas d'utilisation. Pour plus d'informations, consultez politiques gérées par AWS ou politiques gérées par AWS pour les activités professionnelles dans le Guide de l'utilisateur IAM.
- Accordez les autorisations de moindre privilège : lorsque vous définissez des autorisations avec des politiques IAM, accordez uniquement les autorisations nécessaires à l'exécution d'une seule tâche. Pour ce faire, vous définissez les actions qui peuvent être entreprises sur des ressources spécifiques dans des conditions spécifiques, également appelées autorisations de moindre privilège. Pour plus d'informations sur l'utilisation d'IAM pour appliquer des autorisations, consultez politiques et autorisations dans IAM dans le Guide de l'utilisateur IAM.
- Utilisez des conditions dans les politiques IAM pour restreindre davantage l'accès : vous pouvez ajouter une condition à vos politiques afin de limiter l'accès aux actions et aux ressources. Par exemple, vous pouvez écrire une condition de politique pour spécifier que toutes les demandes doivent être envoyées via SSL. Vous pouvez également utiliser des conditions pour accorder l'accès aux actions de service si elles sont utilisées par le biais d'un service spécifique Service AWS, tel que CloudFormation. Pour plus d'informations, consultez Conditions pour éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.
- Utilisez l'Analyseur d'accès IAM pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles : l'Analyseur d'accès IAM valide les politiques nouvelles et existantes de manière à ce que les politiques IAM respectent le langage de politique IAM (JSON) et les bonnes pratiques IAM. IAM Access Analyzer fournit plus de 100 vérifications de politiques et des recommandations exploitables pour vous aider à créer des politiques sécurisées et fonctionnelles. Pour plus d'informations, consultez Validation de politiques avec IAM Access Analyzer dans le Guide de l'utilisateur IAM.
- Exiger l'authentification multifactorielle (MFA) : si vous avez un scénario qui nécessite des utilisateurs IAM ou un utilisateur root, activez l'authentification MFA pour une sécurité accrue.

Compte AWS Pour exiger la MFA lorsque des opérations d'API sont appelées, ajoutez des conditions MFA à vos politiques. Pour plus d'informations, consultez Sécurisation de l'accès aux API avec MFA dans le Guide de l'utilisateur IAM.

Pour plus d'informations sur les bonnes pratiques dans IAM, consultez Bonnes pratiques de sécurité dans IAM dans le Guide de l'utilisateur IAM.

Utilisation de la console AWS PCS

Pour accéder à la console AWS Parallel Computing Service, vous devez disposer d'un ensemble minimal d'autorisations. Ces autorisations doivent vous permettre de répertorier et d'afficher les détails des ressources AWS PCS de votre Compte AWS. Si vous créez une politique basée sur l'identité qui est plus restrictive que l'ensemble minimum d'autorisations requis, la console ne fonctionnera pas comme prévu pour les entités (utilisateurs ou rôles) tributaires de cette politique.

Il n'est pas nécessaire d'accorder des autorisations de console minimales aux utilisateurs qui appellent uniquement l'API AWS CLI ou l' AWS API. Autorisez plutôt l'accès à uniquement aux actions qui correspondent à l'opération d'API qu'ils tentent d'effectuer.

Pour plus d'informations sur les autorisations minimales requises pour utiliser la console AWS PCS, consultezAutorisations minimales pour les AWS PCS.

Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Cet exemple montre comment créer une politique qui permet aux utilisateurs IAM d'afficher les politiques en ligne et gérées attachées à leur identité d'utilisateur. Cette politique inclut les autorisations permettant d'effectuer cette action sur la console ou par programmation à l'aide de l'API AWS CLI or AWS .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
```

```
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS politiques gérées pour le service de calcul AWS parallèle

Une politique AWS gérée est une politique autonome créée et administrée par AWS. AWS les politiques gérées sont conçues pour fournir des autorisations pour de nombreux cas d'utilisation courants afin que vous puissiez commencer à attribuer des autorisations aux utilisateurs, aux groupes et aux rôles.

N'oubliez pas que les politiques AWS gérées peuvent ne pas accorder d'autorisations de moindre privilège pour vos cas d'utilisation spécifiques, car elles sont accessibles à tous les AWS clients. Nous vous recommandons de réduire encore les autorisations en définissant des politiques gérées par le client qui sont propres à vos cas d'utilisation.

Vous ne pouvez pas modifier les autorisations définies dans les politiques AWS gérées. Si les autorisations définies dans une politique AWS gérée sont AWS mises à jour, la mise à jour affecte toutes les identités principales (utilisateurs, groupes et rôles) auxquelles la politique est attachée. AWS est le plus susceptible de mettre à jour une politique AWS gérée lorsqu'une nouvelle politique Service AWS est lancée ou lorsque de nouvelles opérations d'API sont disponibles pour les services existants.

Pour plus d'informations, consultez Politiques gérées par AWS dans le Guide de l'utilisateur IAM.

AWS politique gérée : AWSPCSCompute NodePolicy

Vous pouvez les associer AWSPCSCompute NodePolicy à vos entités IAM. Vous pouvez associer cette politique à un rôle IAM de nœud de calcul AWS PCS que vous spécifiez pour autoriser les nœuds utilisant ce rôle à se connecter à un cluster AWS PCS.

AWS PCS associe cette politique à un rôle de groupe de nœuds de calcul lorsque vous utilisez la console pour créer un groupe de nœuds de calcul.

Détails de l'autorisation

Cette politique inclut les autorisations suivantes.

- `pcs:RegisterComputeNodeGroupInstance`— Autoriser un nœud de calcul AWS PCS (instance EC2) à s'enregistrer auprès d'un cluster AWS PCS.

Pour voir les autorisations de cette stratégie, consultez AWSPCSComputeNodePolicy dans le AWS Guide de référence des stratégies gérées par.

AWS politique gérée : AWSPCSService RolePolicy

Vous ne pouvez pas vous associer AWSPCSService RolePolicy à vos entités IAM. Cette politique est associée à un rôle lié à un service qui permet à AWS PCS d'effectuer des actions en votre nom. Pour de plus amples informations, veuillez consulter Rôles liés aux services pour PCS AWS.

Détails de l'autorisation

Cette politique inclut les autorisations suivantes.

- `ec2`— Permet à AWS PCS de créer et de gérer des ressources Amazon EC2.
- `iam`— Permet à AWS PCS de créer un rôle lié à un service pour le parc Amazon EC2 et de le transmettre à Amazon EC2.
- `cloudwatch`— Permet à AWS PCS de publier des statistiques de service sur Amazon CloudWatch.
- `secretsmanager`— Permet à AWS PCS de gérer les secrets des ressources du cluster AWS PCS.

Pour voir les autorisations de cette stratégie, consultez AWSPCSServiceRolePolicy dans le AWS Guide de référence des stratégies gérées par.

AWS Mises à jour des politiques AWS gérées par PCS

Consultez les détails des mises à jour des politiques AWS gérées pour AWS PCS depuis que ce service a commencé à suivre ces modifications. Pour recevoir des alertes automatiques concernant les modifications apportées à cette page, abonnez-vous au flux RSS sur la page d'historique du document AWS PCS.

Modifier	Description	Date
AWSPCSServiceRolePolicy – Mise à jour d'une stratégie existante	AWS PCS a ajouté de nouvelles autorisations pour prendre en charge les blocs de capacité pour une capacité de calcul prévisible. `ec2:DescribeCapacityReservations` Autorisat	11 septembre 2025

Modifier	Description	Date
	ion ajoutée pour permettre à AWS PCS de découvrir et d'utiliser les réservations de blocs de capacité pour les groupes de nœuds de calcul.	
AWSPCSComputeNodePolicy : nouvelle politique	AWS PCS a ajouté une nouvelle politique autorisant les nœuds de calcul AWS PCS à se connecter aux clusters AWS PCS. AWS PCS associe cette politique à un rôle IAM lorsque vous créez un groupe de nœuds de calcul dans la console AWS PCS.	23 juin 2025
Mise à jour du JSON dans ce document	Le JSON de ce document a été corrigé pour l'inclure `"arn:aws:ec2:*:*:spot-instances-request/*"` .	5 septembre 2024
AWS PCS a commencé à suivre les modifications	AWS PCS a commencé à suivre les modifications apportées AWS à ses politiques gérées.	28 août 2024

Rôles liés aux services pour PCS AWS

AWS Le service de calcul parallèle utilise des Gestion des identités et des accès AWS rôles liés au service (IAM). Un rôle lié à un service est un type unique de rôle IAM directement lié au PCS. AWS Les rôles liés au service sont prédéfinis par AWS PCS et incluent toutes les autorisations dont le service a besoin pour appeler d'autres AWS services en votre nom.

Un rôle lié à un service facilite la configuration de AWS PCS, car vous n'avez pas à ajouter manuellement les autorisations nécessaires. AWS PCS définit les autorisations associées à ses rôles liés aux services et, sauf indication contraire, seul AWS PCS peut assumer ses rôles. Les autorisations définies comprennent la politique de confiance et la politique d'autorisation. De plus, cette politique d'autorisation ne peut pas être attachée à une autre entité IAM.

Vous pouvez supprimer un rôle lié à un service uniquement après la suppression préalable de ses ressources connexes. Cela protège les ressources de votre AWS PC car vous ne pouvez pas supprimer accidentellement l'autorisation d'accès aux ressources.

Pour plus d'informations sur les autres services qui prennent en charge les rôles liés à un service, consultez la section AWS Services qui fonctionnent avec IAM et recherchez les services dont la valeur est Oui dans la colonne Rôles liés à un service. Sélectionnez un Oui ayant un lien pour consulter la documentation du rôle lié à un service, pour ce service.

Autorisations de rôle liées au service pour PCS AWS

AWS PCS utilise le rôle lié à un service nommé AWSServiceRoleForPCS : autorise AWS PCS à gérer les ressources Amazon EC2.

Le rôle lié au service AWSService RoleFor PCS fait confiance aux services suivants pour assumer le rôle :

- `pcs.amazonaws.com`

La politique d'autorisations de rôle nommée AWSPCSServiceRolePolicypermet à AWS PCS d'effectuer des actions sur des ressources spécifiques.

Vous devez configurer les autorisations de manière à permettre à vos utilisateurs, groupes ou rôles de créer, modifier ou supprimer un rôle lié à un service. Pour plus d'informations, consultez Autorisations de rôles liés à un service dans le Guide de l'utilisateur IAM.

Création d'un rôle lié à un service pour PCS AWS

Il n'est pas nécessaire de créer manuellement un rôle lié à un service. AWS PCS crée pour vous un rôle lié à un service lorsque vous créez un cluster.

Modification d'un rôle lié à un service pour PCS AWS

AWS PCS ne vous permet pas de modifier le rôle lié au service AWSService RoleFor PCS. Après avoir créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités

peuvent faire référence à ce rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour plus d'informations, consultez Modification d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Supprimer un rôle lié à un service pour PCS AWS

Si vous n'avez plus besoin d'utiliser une fonctionnalité ou un service qui nécessite un rôle lié à un service, nous vous recommandons de supprimer ce rôle. De cette façon, vous n'avez aucune entité inutilisée qui n'est pas surveillée ou gérée activement. Cependant, vous devez nettoyer les ressources de votre rôle lié à un service avant de pouvoir les supprimer manuellement.

> **Note**
>
> Si le service AWS PCS utilise le rôle lorsque vous essayez de supprimer les ressources, la suppression risque d'échouer. Si cela se produit, patientez quelques minutes et réessayez.

Pour supprimer les ressources AWS PCS utilisées par le AWSService RoleFor PCS

Vous devez supprimer tous vos clusters pour supprimer le rôle lié au service AWSService RoleFor PCS. Pour plus d'informations, voir Supprimer un cluster.

Pour supprimer manuellement le rôle lié au service à l'aide d'IAM

Utilisez la console IAM, le AWS CLI, ou l' AWS API pour supprimer le rôle lié au service AWSService RoleFor PCS. Pour plus d'informations, consultez la section Suppression d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Régions prises en charge pour les AWS rôles liés au service PCS

AWS PCS prend en charge l'utilisation de rôles liés au service dans toutes les régions où le service est disponible. Pour plus d'informations, consultez AWS Régions et points de terminaison.

Rôle Amazon EC2 Spot pour PC AWS

Si vous souhaitez créer un groupe de nœuds de calcul AWS PCS qui utilise Spot comme option d'achat, vous devez également avoir le rôle lié au service AWSServiceRoleForEC2Spot dans votre. Compte AWS Vous pouvez utiliser la AWS CLI commande suivante pour créer le rôle. Pour plus d'informations, voir Créer un rôle lié à un service et Créer un rôle pour déléguer des autorisations à un AWS service dans le Guide de l'Gestion des identités et des accès AWS utilisateur.

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

> **Note**
>
> Le message d'erreur suivant s'affiche si vous Compte AWS possédez déjà un rôle `AWSServiceRoleForEC2Spot` IAM.
>
> ```
> An error occurred (InvalidInput) when calling the CreateServiceLinkedRole
> operation: Service role name AWSServiceRoleForEC2Spot has been taken in this
> account, please try a different suffix.
> ```

Autorisations minimales pour les AWS PCS

Cette section décrit les autorisations IAM minimales requises pour qu'une identité IAM (utilisateur, groupe ou rôle) puisse utiliser le service.

Table des matières

- Autorisations minimales pour utiliser les actions d'API
- Autorisations minimales pour utiliser les balises
- Autorisations minimales pour prendre en charge les journaux
- Autorisations minimales pour utiliser les blocs de capacité
- Autorisations minimales pour un administrateur de service

Autorisations minimales pour utiliser les actions d'API

Action d'API	Autorisations minimales	Autorisations supplémentaires pour la console
CreateCluster	`ec2:CreateNetworkInterface, ec2:DescribeVpcs, ec2:DescribeSubnets,`	

Action d'API	Autorisations minimales	Autorisations supplémentaires pour la console
	`ec2:DescribeSecurityGroups`, `ec2:GetSecurityGroupsForVpc`, `iam:CreateServiceLinkedRole`, `secretsmanager:CreateSecret`, `secretsmanager:TagResource`, `secretsmanager:RotateSecret`, `pcs:CreateCluster`	
ListClusters	`pcs:ListClusters`	
GetCluster	`pcs:GetCluster`	`ec2:DescribeSubnets`
DeleteCluster	`pcs:DeleteCluster`	

Action d'API	Autorisations minimales	Autorisations supplémentaires pour la console
CreateComputeNodeGroup	`ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:DescribeLaunchTemplates, ec2:DescribeLaunchTemplateVersions, ec2:DescribeInstanceTypes, ec2:DescribeInstanceTypeOfferings, ec2:RunInstances, ec2:CreateFleet, ec2:CreateTags, iam:PassRole, iam:GetInstanceProfile, pcs:CreateComputeNodeGroup`	`iam:ListInstanceProfiles, ec2:DescribeImages, pcs:GetCluster`
ListComputerNodeGroups	`pcs:ListComputeNodeGroups`	`pcs:GetCluster`
GetComputeNodeGroup	`pcs:GetComputeNodeGroup`	`ec2:DescribeSubnets`

Action d'API	Autorisations minimales	Autorisations supplémentaires pour la console
UpdateComputeNodeGroup	`ec2:DescribeVpcs, ec2:DescribeSubnets, ec2:DescribeSecurityGroups, ec2:DescribeLaunchTemplates, ec2:DescribeLaunchTemplateVersions, ec2:DescribeInstanceTypes, ec2:DescribeInstanceTypeOfferings, ec2:RunInstances, ec2:CreateFleet, ec2:CreateTags, iam:PassRole, iam:GetInstanceProfile, pcs:UpdateComputeNodeGroup`	`pcs:GetComputeNodeGroup, iam:ListInstanceProfiles, ec2:DescribeImages, pcs:GetCluster`
DeleteComputeNodeGroup	`pcs:DeleteComputeNodeGroup`	
CreateQueue	`pcs:CreateQueue`	`pcs:ListComputeNodeGroups, pcs:GetCluster`
ListQueues	`pcs:ListQueues`	`pcs:GetCluster`
GetQueue	`pcs:GetQueue`	

Action d'API	Autorisations minimales	Autorisations supplémentaires pour la console
UpdateQueue	`pcs:UpdateQueue`	`pcs:ListComputeNodeGroups, pcs:GetQueue`
DeleteQueue	`pcs:DeleteQueue`	

Autorisations minimales pour utiliser les balises

Les autorisations suivantes sont requises pour utiliser des balises avec vos ressources dans AWS PCS.

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

Autorisations minimales pour prendre en charge les journaux

AWS PCS envoie les données du journal à Amazon CloudWatch Logs (CloudWatch Logs). Vous devez vous assurer que votre identité dispose des autorisations minimales pour utiliser CloudWatch les journaux. Pour plus d'informations, consultez la section Présentation de la gestion des autorisations d'accès à vos ressources CloudWatch Logs dans le guide de l'utilisateur Amazon CloudWatch Logs.

Pour plus d'informations sur les autorisations requises pour qu'un service envoie des CloudWatch journaux à Logs, consultez la section Activation de la journalisation à partir AWS des services dans le guide de l'utilisateur Amazon CloudWatch Logs.

Autorisations minimales pour utiliser les blocs de capacité

Amazon EC2 Capacity Blocks for ML est une option d'achat Amazon EC2 qui vous permet de payer à l'avance pour réserver des instances de calcul accéléré basées sur un GPU à une date et à une heure spécifiques afin de prendre en charge des charges de travail de courte durée. Pour de plus amples informations, veuillez consulter Utilisation des blocs de capacité Amazon EC2 pour le ML avec PCS AWS.

Vous choisissez d'utiliser des blocs de capacité lorsque vous créez ou mettez à jour un groupe de nœuds de calcul. L'identité IAM que vous utilisez pour créer ou mettre à jour le groupe de nœuds de calcul doit disposer de l'autorisation suivante :

```
ec2:DescribeCapacityReservations
```

Autorisations minimales pour un administrateur de service

La politique IAM suivante spécifie les autorisations minimales requises pour une identité IAM (utilisateur, groupe ou rôle) afin de configurer et de gérer le service AWS PCS.

> **Note**
>
> Les utilisateurs qui ne configurent ni ne gèrent le service n'ont pas besoin de ces autorisations. Les utilisateurs qui exécutent uniquement des tâches utilisent le protocole Secure Shell (SSH) pour se connecter au cluster. Gestion des identités et des accès AWS (IAM) ne gère pas l'authentification ou l'autorisation pour SSH.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PCSAccess",
      "Effect": "Allow",
      "Action": [
        "pcs:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "EC2Access",
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface",
        "ec2:DescribeImages",
        "ec2:GetSecurityGroupsForVpc",
        "ec2:DescribeSubnets",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeVpcs",
        "ec2:DescribeLaunchTemplates",
```

```
        "ec2:DescribeLaunchTemplateVersions",
        "ec2:DescribeInstanceTypes",
        "ec2:DescribeInstanceTypeOfferings",
        "ec2:RunInstances",
        "ec2:CreateFleet",
        "ec2:CreateTags",
        "ec2:DescribeCapacityReservations"
      ],
      "Resource": "*"
    },
    {
      "Sid": "IamInstanceProfile",
      "Effect": "Allow",
      "Action": [
        "iam:GetInstanceProfile"
      ],
      "Resource": "*"
    },
    {
      "Sid": "IamPassRole",
      "Effect": "Allow",
      "Action": [
        "iam:PassRole"
      ],
      "Resource": [
        "arn:aws:iam::*:role/*/AWSPCS*",
        "arn:aws:iam::*:role/AWSPCS*",
        "arn:aws:iam::*:role/aws-pcs/*",
        "arn:aws:iam::*:role/*/aws-pcs/*"
      ],
      "Condition": {
        "StringEquals": {
           "iam:PassedToService": [
             "ec2.amazonaws.com"
           ]
        }
      }
    },
    {
      "Sid": "SLRAccess",
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole"
      ],
```

```
      "Resource": [
        "arn:aws:iam::*:role/aws-service-role/pcs.amazonaws.com/AWSServiceRoleFor*",
        "arn:aws:iam::*:role/aws-service-role/spot.amazonaws.com/AWSServiceRoleFor*"
      ],
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": [
            "pcs.amazonaws.com",
            "spot.amazonaws.com"
          ]
        }
      }
    },
    {
      "Sid": "AccessKMSKey",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:GenerateDataKey",
        "kms:CreateGrant",
        "kms:DescribeKey"
      ],
      "Resource": "*"
    },
    {
      "Sid": "SecretManagementAccess",
      "Effect": "Allow",
      "Action": [
        "secretsmanager:CreateSecret",
        "secretsmanager:TagResource",
        "secretsmanager:UpdateSecret",
        "secretsmanager:RotateSecret"
      ],
      "Resource": "*"
    },
    {
       "Sid": "ServiceLogsDelivery",
       "Effect": "Allow",
       "Action": [
         "pcs:AllowVendedLogDeliveryForResource",
         "logs:PutDeliverySource",
         "logs:PutDeliveryDestination",
         "logs:CreateDelivery"
```

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

Profils d'instance IAM pour AWS Parallel Computing Service

Les applications qui s'exécutent sur une instance EC2 doivent inclure des AWS informations d'identification dans toutes les demandes d' AWS API qu'elles effectuent. Nous vous recommandons d'utiliser un rôle IAM pour gérer les informations d'identification temporaires sur l'instance EC2. Pour ce faire, vous pouvez définir un profil d'instance et l'associer à vos instances. Pour plus d'informations, consultez la section Rôles IAM pour Amazon EC2 dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

> **Note**
>
> Lorsque vous utilisez le AWS Management Console pour créer un rôle IAM pour Amazon EC2, la console crée automatiquement un profil d'instance et lui donne le même nom que le rôle IAM. Si vous utilisez les AWS CLI actions d' AWS API ou un AWS SDK pour créer le rôle IAM, vous créez le profil d'instance en tant qu'action distincte. Pour plus d'informations, consultez la section Profils d'instance dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Vous devez spécifier le nom de ressource Amazon (ARN) d'un profil d'instance lorsque vous créez un groupe de nœuds de calcul. Vous pouvez choisir différents profils d'instance pour certains ou pour tous les groupes de nœuds de calcul.

Exigences

Rôle IAM du profil d'instance

Le rôle IAM associé au profil d'instance doit figurer `/aws-pcs/` dans son chemin, ou son nom doit commencer `AWSPCS` par.

Exemple de rôle IAM ARNs

- `arn:aws:iam::*:role/AWSPCS-example-role-1`
- `arn:aws:iam::*:role/aws-pcs/example-role-2`

Permissions

Le rôle IAM associé au profil d'instance pour AWS PCS doit inclure la politique suivante.

JSON

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

Politiques supplémentaires

Envisagez d'ajouter des politiques gérées au profil d'instance. Par exemple :

- AmazonS3 ReadOnlyAccess fournit un accès en lecture seule à tous les compartiments S3.
- Amazon SSMManaged InstanceCore active les fonctionnalités principales du service AWS Systems Manager, telles que l'accès à distance directement depuis Amazon Management Console.
- CloudWatchAgentServerPolicycontient les autorisations requises pour une utilisation AmazonCloudWatchAgent sur les serveurs.

Vous pouvez également inclure vos propres politiques IAM adaptées à votre cas d'utilisation spécifique.

Création d'un profil d'instance pour AWS PCS

AWS PCS console

Sélectionnez Créer un profil de base lorsque vous créez un groupe de nœuds de calcul pour que AWS PCS en crée un pour vous avec la politique minimale requise.

Amazon EC2 console

Vous pouvez créer un profil d'instance directement depuis la console Amazon EC2. Pour plus d'informations, consultez la section Utilisation des profils d'instance dans le Guide de Gestion des identités et des accès AWS l'utilisateur.

> **Important**
>
> Assurez-vous d'utiliser le préfixe requis `AWSPCS` dans le nom du rôle IAM.

AWS CLI

Configuration du profil d'instance de base à l'aide de l'AWS CLI

> **Note**
>
> Remplacez *`example-role`* dans les exemples suivants par le nom de votre rôle IAM.

1. Créez un rôle IAM avec `/aws-pcs/` comme attribut de chemin ou un nom commençant `AWSPCS` par.

   a. Copiez et collez le contenu suivant dans un nouveau fichier texte nommé`trust_policy.json`.

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
             "Principal": {
                "Service": [
                    "ec2.amazonaws.com"
                ]
            },
            "Action": [
                "sts:AssumeRole"
            ]
        }
```

```
    ]
}
```

b. Utilisez l'une des commandes suivantes pour créer le rôle IAM.

```
aws iam create-role --path /aws-pcs/ --role-name example-role --assume-role-
policy-document file://trust_policy.json
```

or

```
aws iam create-role --role-name AWSPCS-example-role --assume-role-policy-
document file://trust_policy.json
```

2. Joignez des autorisations.

a. Copiez et collez le contenu suivant dans un nouveau fichier texte nommé`policy_document.json`.

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

b. Joignez le document de politique au rôle. Cette commande associe la politique en tant que politique en ligne.

```
aws iam put-role-policy \
    --role-name example-role \
    --policy-name pcsRegisterInstancePolicy \
    --policy-document file://policy_document.json
```

3. Créez un profil d'instance. ***example-profile***Remplacez-le par le nom de votre profil d'instance.

```
aws iam create-instance-profile --instance-profile-name example-profile
```

4. Associez le rôle IAM au profil d'instance.

```
aws iam add-role-to-instance-profile \
   --instance-profile-name example-profile \
   --role-name example-role
```

Rechercher les profils d'instance utilisés avec AWS PCS

1. Si vous ne connaissez pas le nom exact de vos rôles IAM pour AWS PCS, utilisez la AWS CLI commande suivante pour répertorier les rôles IAM qui répondent aux exigences relatives aux noms AWS PCS.

```
aws iam list-roles --query "Roles[?starts_with(RoleName, 'AWSPCS') ||
 contains(Path, '/aws-pcs/')].[RoleName]" --output text
```

2. Utilisez la AWS CLI commande suivante pour répertorier les profils d'instance associés à un rôle IAM spécifique. *role-name*Remplacez-le par le nom d'un rôle IAM répondant aux exigences du AWS PCS en matière de nom.

```
aws iam list-instance-profiles-for-role --role-name role-name
```

Résolution des problèmes d'identité et d'accès au service de calcul AWS parallèle

Utilisez les informations suivantes pour vous aider à diagnostiquer et à résoudre les problèmes courants que vous pouvez rencontrer lorsque vous travaillez avec AWS PCS et IAM.

Rubriques

- Je ne suis pas autorisé à effectuer une action dans AWS PCS
- Je ne suis pas autorisé à effectuer iam : PassRole
- Je souhaite autoriser des personnes extérieures à moi Compte AWS à accéder aux ressources de mon AWS PCS

Je ne suis pas autorisé à effectuer une action dans AWS PCS

Si vous recevez une erreur qui indique que vous n'êtes pas autorisé à effectuer une action, vos politiques doivent être mises à jour afin de vous permettre d'effectuer l'action.

L'exemple d'erreur suivant se produit quand l'utilisateur IAM `mateojackson` tente d'utiliser la console pour afficher des informations détaillées sur une ressource *`my-example-widget`* fictive, mais ne dispose pas des autorisations `pcs:`*`GetWidget`* fictives.

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

Dans ce cas, la politique qui s'applique à l'utilisateur `mateojackson` doit être mise à jour pour autoriser l'accès à la ressource *`my-example-widget`* à l'aide de l'action `pcs:`*`GetWidget`*.

Si vous avez besoin d'aide, contactez votre AWS administrateur. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je ne suis pas autorisé à effectuer iam : PassRole

Si vous recevez un message d'erreur indiquant que vous n'êtes pas autorisé à effectuer l'`iam:PassRole`action, vos politiques doivent être mises à jour pour vous permettre de transmettre un rôle à AWS PCS.

Certains vous Services AWS permettent de transmettre un rôle existant à ce service au lieu de créer un nouveau rôle de service ou un rôle lié à un service. Pour ce faire, vous devez disposer des autorisations nécessaires pour transmettre le rôle au service.

L'exemple d'erreur suivant se produit lorsqu'un utilisateur IAM nommé `marymajor` essaie d'utiliser la console pour effectuer une action dans AWS PCS. Toutefois, l'action nécessite que le service ait des autorisations accordées par un rôle de service. Mary n'est pas autorisée à transmettre le rôle au service.

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

Dans ce cas, les politiques de Mary doivent être mises à jour pour lui permettre d'exécuter l'action `iam:PassRole`.

Si vous avez besoin d'aide, contactez votre AWS administrateur. Votre administrateur vous a fourni vos informations d'identification de connexion.

Je souhaite autoriser des personnes extérieures à moi Compte AWS à accéder aux ressources de mon AWS PCS

Vous pouvez créer un rôle que les utilisateurs provenant d'autres comptes ou les personnes extérieures à votre organisation pourront utiliser pour accéder à vos ressources. Vous pouvez spécifier qui est autorisé à assumer le rôle. Pour les services qui prennent en charge les politiques basées sur les ressources ou les listes de contrôle d'accès (ACLs), vous pouvez utiliser ces politiques pour autoriser les utilisateurs à accéder à vos ressources.

Pour plus d'informations, consultez les éléments suivants :

- Pour savoir si AWS PCS prend en charge ces fonctionnalités, consultezComment fonctionne AWS Parallel Computing Service avec IAM.
- Pour savoir comment fournir l'accès à vos ressources sur celles Comptes AWS que vous possédez, consultez la section Fournir l'accès à un utilisateur IAM dans un autre utilisateur Compte AWS que vous possédez dans le Guide de l'utilisateur IAM.
- Pour savoir comment fournir l'accès à vos ressources à des tiers Comptes AWS, consultez la section Fournir un accès à des ressources Comptes AWS détenues par des tiers dans le guide de l'utilisateur IAM.
- Pour savoir comment fournir un accès par le biais de la fédération d'identité, consultez Fournir un accès à des utilisateurs authentifiés en externe (fédération d'identité) dans le Guide de l'utilisateur IAM.
- Pour en savoir plus sur la différence entre l'utilisation des rôles et des politiques basées sur les ressources pour l'accès intercompte, consultez Accès intercompte aux ressources dans IAM dans le Guide de l'utilisateur IAM.

Validation de conformité pour le service de calcul AWS parallèle

Pour savoir si un programme Services AWS de conformité Service AWS s'inscrit dans le champ d'application de programmes de conformité spécifiques, consultez Services AWS la section de conformité et sélectionnez le programme de conformité qui vous intéresse. Pour des informations générales, voir Programmes de AWS conformité Programmes AWS de .

Vous pouvez télécharger des rapports d'audit tiers à l'aide de AWS Artifact. Pour plus d'informations, voir Téléchargement de rapports dans AWS Artifact .

Votre responsabilité en matière de conformité lors de l'utilisation Services AWS est déterminée par la sensibilité de vos données, les objectifs de conformité de votre entreprise et les lois et réglementations applicables. Pour plus d'informations sur votre responsabilité en matière de conformité lors de l'utilisation Services AWS, consultez AWS la documentation de sécurité.

Résilience dans les services de calcul AWS parallèle

L'infrastructure AWS mondiale est construite autour Régions AWS de zones de disponibilité. Régions AWS fournissent plusieurs zones de disponibilité physiquement séparées et isolées, connectées par un réseau à faible latence, à haut débit et hautement redondant. Avec les zones de disponibilité, vous pouvez concevoir et exploiter des applications et des bases de données qui basculent automatiquement d'une zone à l'autre sans interruption. Les zones de disponibilité sont davantage disponibles, tolérantes aux pannes et ont une plus grande capacité de mise à l'échelle que les infrastructures traditionnelles à un ou plusieurs centres de données.

Pour plus d'informations sur les zones de disponibilité Régions AWS et les zones de disponibilité, consultez la section Infrastructure AWS globale.

Sécurité de l'infrastructure dans un service de calcul AWS parallèle

En tant que service géré, AWS Parallel Computing Service est protégé par la sécurité du réseau AWS mondial. Pour plus d'informations sur les services AWS de sécurité et sur la manière dont AWS l'infrastructure est protégée, consultez la section Sécurité du AWS cloud. Pour concevoir votre AWS environnement en utilisant les meilleures pratiques en matière de sécurité de l'infrastructure, consultez la section Protection de l'infrastructure dans le cadre AWS bien architecturé du pilier de sécurité.

Vous utilisez des appels d'API AWS publiés pour accéder à AWS PCS via le réseau. Les clients doivent prendre en charge les éléments suivants :

- Protocole TLS (Transport Layer Security). Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Ses suites de chiffrement PFS (Perfect Forward Secrecy) comme DHE (Ephemeral Diffie-Hellman) ou ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes.

Lorsque AWS PCS crée un cluster, le service lance le contrôleur Slurm dans un compte appartenant au service, distinct des nœuds de calcul de votre compte. Pour établir une passerelle entre le contrôleur et les nœuds de calcul, AWS PCS crée une interface réseau élastique (ENI) inter-comptes

dans votre VPC. Le contrôleur Slurm utilise l'ENI pour gérer et communiquer avec les nœuds de calcul à travers différents nœuds Comptes AWS, en maintenant la sécurité et l'isolation des ressources tout en facilitant l'efficacité du HPC et des opérations. AI/ML

Analyse et gestion des vulnérabilités dans le service de calcul AWS parallèle

La configuration et les contrôles informatiques sont une responsabilité partagée entre vous AWS et vous. Pour plus d'informations, consultez le modèle de responsabilitéAWS partagée. AWS gère les tâches de sécurité de base pour l'infrastructure sous-jacente du compte de service, telles que l'application de correctifs au système d'exploitation sur les instances de contrôleur, la configuration du pare-feu et la reprise après sinistre de AWS l'infrastructure. Ces procédures ont été vérifiées et certifiées par les tiers appropriés. Pour plus de détails, consultez les meilleures pratiques en matière de sécurité, d'identité et de conformité.

> **Note**
>
> Les manettes Slurm ne sont pas disponibles pendant leur mise à jour. Les tâches en cours ne sont pas affectées. Les tâches soumises lorsque le contrôleur du cluster n'est pas disponible sont conservées jusqu'à ce que le contrôleur soit disponible.

Vous êtes responsable de la sécurité de l'infrastructure sous-jacente dans votre Compte AWS :

- Maintenez votre code, y compris les mises à jour et les correctifs de sécurité.
- Corrigez et mettez à jour le système d'exploitation dans Amazon Machine Image (AMI) pour vos groupes de nœuds de calcul et mettez à jour vos groupes de nœuds de calcul pour utiliser l'AMI mise à jour.
- Mettez à jour le planificateur pour qu'il reste dans les versions prises en charge. Mettez à jour l'AMI pour vos groupes de nœuds de calcul et mettez à jour votre groupe de nœuds de calcul pour utiliser l'AMI mise à jour.
- Authentifiez et chiffrez les communications entre les clients utilisateurs et les nœuds auxquels ils se connectent.

Pour plus d'informations sur la mise à jour de l'AMI pour vos groupes de nœuds de calcul, consultezAmazon Machine Images (AMIs) pour AWS PC.

Prévention du problème de l'adjoint confus entre services

Le problème de député confus est un problème de sécurité dans lequel une entité qui n'a pas l'autorisation d'effectuer une action peut contraindre une entité plus privilégiée à effectuer cette action. En AWS, l'usurpation d'identité interservices peut entraîner la confusion des adjoints. L'usurpation d'identité entre services peut se produire lorsqu'un service (le service appelant) appelle un autre service (le service appelé). Le service appelant peut être manipulé et ses autorisations utilisées pour agir sur les ressources d'un autre client auxquelles on ne serait pas autorisé à accéder autrement. Pour éviter cela, AWS fournit des outils qui vous aident à protéger vos données pour tous les services avec des principaux de service qui ont eu accès aux ressources de votre compte.

Nous recommandons d'utiliser les clés de contexte de condition `aws:SourceAccount`globale `aws:SourceArn`et les clés contextuelles dans les politiques de ressources afin de limiter les autorisations que le AWS Parallel Computing Service (AWS PCS) accorde à un autre service à la ressource. Utilisez `aws:SourceArn` si vous souhaitez qu'une seule ressource soit associée à l'accès entre services. Utilisez `aws:SourceAccount` si vous souhaitez autoriser l'association d'une ressource de ce compte à l'utilisation interservices.

Le moyen le plus efficace de se protéger contre le problème de député confus consiste à utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'ARN complet de la ressource. Si vous ne connaissez pas l'ARN complet de la ressource ou si vous spécifiez plusieurs ressources, utilisez la clé de contexte de condition globale `aws:SourceArn` avec des caractères génériques (*) pour les parties inconnues de l'ARN. Par exemple, `arn:aws:`*`servicename`*`:*:`*`123456789012`*`:*`.

Si la valeur `aws:SourceArn` ne contient pas l'ID du compte, tel qu'un ARN de compartiment Amazon S3, vous devez utiliser les deux clés de contexte de condition globale pour limiter les autorisations.

La valeur de `aws:SourceArn` doit être un ARN de cluster.

L'exemple suivant montre comment vous pouvez utiliser les touches `aws:SourceArn` contextuelles et de condition `aws:SourceAccount` globale dans AWS PCS pour éviter le problème de confusion des adjoints.

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

Rôle IAM pour les instances Amazon EC2 mises en service dans le cadre d'un groupe de nœuds de calcul

AWS PCS orchestre automatiquement la capacité Amazon EC2 pour chacun des groupes de nœuds de calcul configurés dans un cluster. Lors de la création d'un groupe de nœuds de calcul, les utilisateurs doivent fournir un profil d'instance IAM via le `iamInstanceProfileArn` champ. Le profil d'instance spécifie les autorisations associées aux instances EC2 provisionnées. AWS PCS accepte tout rôle ayant `AWSPCS` comme préfixe de nom de `/aws-pcs/` rôle ou faisant partie du chemin du rôle. L'`iam:PassRole`autorisation est requise pour l'identité IAM (utilisateur ou rôle) qui crée ou met à jour un groupe de nœuds de calcul. Lorsqu'un utilisateur appelle les actions de `UpdateComputeNodeGroup` l'API `CreateComputeNodeGroup` or, AWS PCS vérifie si l'utilisateur est autorisé à effectuer l'`iam:PassRole`action.

L'exemple de politique suivant accorde les autorisations de transmettre uniquement des rôles IAM dont le nom commence par `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"
                    ]
                }
            }
        }
    ]
}
```

Bonnes pratiques de sécurité pour le service de calcul AWS parallèle

Cette section décrit les meilleures pratiques de sécurité spécifiques au AWS Parallel Computing Service (AWS PCS). Pour en savoir plus sur les meilleures pratiques en matière de sécurité dans AWS, consultez la section Meilleures pratiques en matière de sécurité, d'identité et de conformité.

Sécurité liée à l'AMI

- N'utilisez pas d'échantillon AWS PCS AMIs pour les charges de travail de production. Les échantillons AMIs ne sont pas pris en charge et sont uniquement destinés à être testés.
- Mettez régulièrement à jour le système d'exploitation et le logiciel de l'AMI pour vos groupes de nœuds de calcul afin d'atténuer les vulnérabilités.
- N'utilisez que des packages AWS PCS officiels authentifiés téléchargés à partir de AWS sources officielles.
- Mettez régulièrement à jour les packages AWS PCS dans l'AMI pour les groupes de nœuds de calcul et mettez à jour les nœuds de calcul pour utiliser l'AMI mise à jour. Envisagez d'automatiser ce processus afin de minimiser les vulnérabilités.

Pour de plus amples informations, veuillez consulter Images Amazon Machine personnalisées (AMIs) pour AWS PC.

Sécurité de Slurm Workload Manager

- Mettez en œuvre des contrôles d'accès et des restrictions réseau pour sécuriser les nœuds de contrôle et de calcul de Slurm. Autorisez uniquement les utilisateurs et les systèmes fiables à soumettre des tâches et à accéder aux commandes de gestion de Slurm.

- Utilisez les fonctionnalités de sécurité intégrées de Slurm, telles que l'authentification Slurm, pour vous assurer que les soumissions de tâches et les communications sont authentifiées.
- Mettez à jour les versions de Slurm pour garantir le bon fonctionnement des opérations et la prise en charge des clusters.

  > **Important**
  >
  > Tout cluster qui utilise une version de Slurm ayant atteint la fin de vie du support (EOSL) est immédiatement arrêté. Utilisez le lien en haut des pages du guide de l'utilisateur pour vous abonner au flux RSS de la documentation AWS PCS afin de recevoir une notification lorsqu'une version de Slurm approche d'EOSL.

  Pour de plus amples informations, veuillez consulter Versions Slurm en PCS AWS.
- Effectuez régulièrement une rotation des secrets de cluster afin de garantir la conformité en matière de sécurité et de remédier aux risques de sécurité potentiels. Cela est nécessaire pour se conformer aux normes HIPAA et FedRAMP.

  Pour de plus amples informations, veuillez consulter Rotation des secrets de cluster dans AWS PCS.

Surveillance et journalisation

- Utilisez Amazon CloudWatch Logs et AWS CloudTrail pour surveiller et enregistrer les actions dans vos clusters et Compte AWS. Utilisez les données pour le dépannage et l'audit.

Sécurité du réseau

- Déployez vos clusters AWS PCS dans un VPC distinct pour isoler votre environnement HPC du reste du trafic réseau.
- Utilisez des groupes de sécurité et des listes de contrôle d'accès réseau (ACLs) pour contrôler le trafic entrant et sortant vers les instances et les sous-réseaux AWS PCS.
- Utilisez AWS PrivateLink des points de terminaison VPC pour maintenir le trafic réseau entre vos clusters et les autres AWS services du réseau. AWS Pour de plus amples informations, veuillez consulter Accès AWS Parallel Computing Service via un point de terminaison d'interface (AWS PrivateLink).

Journalisation et surveillance pour AWS PCS

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité et des performances de AWS PCS et de vos autres ressources AWS. AWS fournit les outils de surveillance suivants pour surveiller les AWS PCS, signaler un problème et prendre des mesures automatiques le cas échéant :

- Amazon CloudWatch surveille vos AWS ressources et les applications que vous utilisez AWS en temps réel. Vous pouvez collecter et suivre les métriques, créer des tableaux de bord personnalisés, et définir des alarmes qui vous informent ou prennent des mesures lorsqu'une métrique spécifique atteint un seuil que vous spécifiez. Par exemple, vous pouvez CloudWatch suivre l'utilisation du processeur ou d'autres indicateurs de vos instances Amazon EC2 et lancer automatiquement de nouvelles instances en cas de besoin. Pour plus d'informations, consultez le guide de CloudWatch l'utilisateur Amazon.
- Amazon CloudWatch Logs vous permet de surveiller, de stocker et d'accéder à vos fichiers journaux à partir d'instances Amazon EC2 et d'autres sources. CloudTrail CloudWatch Les journaux peuvent surveiller les informations contenues dans les fichiers journaux et vous avertir lorsque certains seuils sont atteints. Vous pouvez également archiver vos données de journaux dans une solution de stockage hautement durable. Pour plus d'informations, consultez le guide de l'utilisateur d'Amazon CloudWatch Logs.
- AWS CloudTrailcapture les appels d'API et les événements associés effectués par ou pour le compte de votre AWS compte et envoie les fichiers journaux dans un compartiment Amazon S3 que vous spécifiez. Vous pouvez identifier les utilisateurs et les comptes appelés AWS, l'adresse IP source à partir de laquelle les appels ont été effectués et la date des appels. Pour plus d'informations, consultez le Guide de l'utilisateur AWS CloudTrail.

Journaux d'achèvement des tâches dans AWS PCS

Les journaux d'achèvement des tâches vous fournissent des informations clés sur vos tâches du AWS Parallel Computing Service (AWS PCS) lorsqu'elles sont terminées, sans frais supplémentaires. Vous pouvez utiliser d'autres AWS services pour accéder à vos données de journal et les traiter, tels qu'Amazon CloudWatch Logs, Amazon Simple Storage Service (Amazon S3) et Amazon Data Firehose AWS ; PCS enregistre les métadonnées relatives à vos tâches, telles que les suivantes.

- ID et nom du job
- Informations sur les utilisateurs et les groupes

- État du poste (tel que`COMPLETED`,`FAILED`,`CANCELLED`)
- Partition utilisée
- Limites de temps
- Heures de début, de fin, de soumission et heures d'éligibilité
- Liste et nombre de nœuds
- Nombre de processeurs
- Répertoire de travail
- Utilisation des ressources (processeur, mémoire)
- Codes de sortie
- Détails des nœuds (noms, instances IDs, types d'instances)

Table des matières

- Conditions préalables
- Configurer les journaux d'achèvement des tâches
- Comment trouver les journaux d'achèvement des tâches
  - CloudWatch Journaux
  - Amazon S3
- Champs du journal d'achèvement des tâches
- Exemples de journaux d'achèvement des tâches

Conditions préalables

Le principal IAM qui gère le cluster AWS PCS doit autoriser l'`pcs:AllowVendedLogDeliveryForResource`action.

L'exemple de politique IAM suivant accorde les autorisations requises.

JSON

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

```
            "Sid": "PcsAllowVendedLogsDelivery",
            "Effect": "Allow",
            "Action": ["pcs:AllowVendedLogDeliveryForResource"],
            "Resource": [
               "arn:aws:pcs:*::cluster/*"
            ]
         }
      ]
}
```

Configurer les journaux d'achèvement des tâches

Vous pouvez configurer des journaux d'achèvement des tâches pour votre cluster AWS PCS à l'aide du AWS Management Console ou AWS CLI.

AWS Management Console

Pour configurer les journaux d'achèvement des tâches à l'aide de la console

1. Ouvrez la console AWS PCS.
2. Dans le panneau de navigation, choisissez Clusters.
3. Choisissez le cluster dans lequel vous souhaitez ajouter les journaux d'achèvement des tâches.
4. Sur la page des détails du cluster, choisissez l'onglet Logs.
5. Sous Job Completion Logs, choisissez Ajouter pour ajouter jusqu'à 3 destinations de livraison de CloudWatch journaux parmi Logs, Amazon S3 et Firehose.
6. Choisissez Mettre à jour les livraisons de journaux.

AWS CLI

Pour configurer les journaux d'achèvement des tâches à l'aide du AWS CLI

1. Créez une destination de livraison du journal :

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

Remplacez :

- *region*— L' Région AWS endroit où vous souhaitez créer la destination, tel que `us-east-1`
- *pcs-logs-destination*— Un nom pour la destination
- *resource-arn*— Le nom de ressource Amazon (ARN) d'un groupe de CloudWatch journaux Logs, d'un bucket S3 ou d'un flux de diffusion Firehose.

Pour plus d'informations, consultez PutDeliveryDestinationle manuel Amazon CloudWatch Logs API Reference.

2. Définissez le cluster PCS comme source de livraison de journaux :

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

Remplacez :

- *region*— Le Région AWS nom de votre cluster, tel que `us-east-1`
- *cluster-logs-source-name*— Un nom pour la source
- *cluster-arn*— l'ARN de votre cluster AWS PCS

Pour plus d'informations, consultez PutDeliverySourcele manuel Amazon CloudWatch Logs API Reference.

3. Connectez la source de livraison à la destination de livraison :

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

Remplacez :

- *region*— Le Région AWS, tel que `us-east-1`
- *cluster-logs-source*— Le nom de votre source de livraison

- *destination-arn*— L'ARN de votre destination de livraison

Pour plus d'informations, consultez CreateDeliveryle manuel Amazon CloudWatch Logs API Reference.

Comment trouver les journaux d'achèvement des tâches

Vous pouvez configurer les destinations des CloudWatch journaux dans Logs et Amazon S3. AWS PCS utilise les noms de chemin et de fichier structurés suivants.

CloudWatch Journaux

AWS PCS utilise le format de nom suivant pour le flux CloudWatch Logs :

```
AWSLogs/PCS/cluster-id/jobcomp.log
```

Par exemple : `AWSLogs/PCS/pcs_abc123de45/jobcomp.log`

Amazon S3

AWS PCS utilise le format de nom suivant pour le chemin S3 :

```
AWSLogs/account-id/PCS/region/cluster-id/jobcomp/year/month/day/hour/
```

Par exemple : `AWSLogs/111122223333/PCS/us-east-1/pcs_abc123de45/jobcomp/2025/06/19/11/`

AWS PCS utilise le format de nom suivant pour les fichiers journaux :

```
PCS_jobcomp_year-month-day-hour_cluster-id_random-id.log.gz
```

Par exemple : `PCS_jobcomp_2025-06-19-11_pcs_abc123de45_04be080b.log.gz`

Champs du journal d'achèvement des tâches

AWS PCS écrit les données du journal d'achèvement des tâches sous forme d'objets JSON. Le conteneur JSON `jobcomp` contient les détails de la tâche. Le tableau suivant décrit les champs à l'intérieur du `jobcomp` conteneur. Certains champs ne sont présents que dans des circonstances spécifiques, par exemple pour les tâches de matrice ou les tâches hétérogènes.

Champs du journal d'achèvement des tâches

Nom	Exemple de valeur	Obligatoire	Remarques
`job_id`	`11`	oui	Toujours présent avec de la valeur
`user`	`"root"`	oui	Toujours présent avec de la valeur
`user_id`	`0`	oui	Toujours présent avec de la valeur
`group`	`"root"`	oui	Toujours présent avec de la valeur
`group_id`	`0`	oui	Toujours présent avec de la valeur
`name`	`"wrap"`	oui	Toujours présent avec de la valeur
`job_state`	`"COMPLETED"`	oui	Toujours présent avec de la valeur
`partition`	`"Hydra-MpiQueue-abcdef01-7"`	oui	Toujours présent avec de la valeur
`time_limit`	`"UNLIMITED"`	oui	Toujours présent, mais peut-être `"UNLIMITED"`
`start_time`	`"2025-06-19T10:58:57"`	oui	Toujours présent, mais peut-être `"Unknown"`
`end_time`	`"2025-06-19T10:58:57"`	oui	Toujours présent, mais peut-être `"Unknown"`
`node_list`	`"Hydra-MpiNG-abcdef01-2345-1"`	oui	Toujours présent avec de la valeur
`node_cnt`	`1`	oui	Toujours présent avec de la valeur

Nom	Exemple de valeur	Obligatoire	Remarques
`proc_cnt`	`1`	oui	Toujours présent avec de la valeur
`work_dir`	`"/root"`	oui	Toujours présent, mais peut-être `"Unknown"`
`reservation_name`	`"weekly_maintenance"`	oui	Toujours présent, mais il peut s'agir d'une chaîne vide `""`
`tres.cpu`	`1`	oui	Toujours présent avec de la valeur
`tres.mem.val`	`600`	oui	Toujours présent avec de la valeur
`tres.mem.unit`	`"M"`	oui	Peut être `"M"` ou `"bb"`
`tres.node`	`1`	oui	Toujours présent avec de la valeur
`tres.billing`	`1`	oui	Toujours présent avec de la valeur
`account`	`"finance"`	oui	Toujours présent, mais il peut s'agir d'une chaîne vide `""`
`qos`	`"normal"`	oui	Toujours présent, mais il peut s'agir d'une chaîne vide `""`
`wc_key`	`"project_1"`	oui	Toujours présent, mais il peut s'agir d'une chaîne vide `""`
`cluster`	`"unknown"`	oui	Toujours présent, mais peut-être `"unknown"`
`submit_time`	`"2025-06-19T10:55:46"`	oui	Toujours présent, mais peut-être `"Unknown"`

Nom	Exemple de valeur	Obligatoire	Remarques
`eligible_time`	`"2025-06-19T10:55:46"`	oui	Toujours présent, mais peut-être `"Unknown"`
`array_job_id`	`12`	non	Présent uniquement si la tâche est une tâche matricielle
`array_task_id`	`1`	non	Présent uniquement si la tâche est une tâche matricielle
`het_job_id`	`10`	non	Présent uniquement s'il s'agit d'une tâche hétérogène
`het_job_offset`	`0`	non	Présent uniquement s'il s'agit d'une tâche hétérogène
`derived_exit_code_status`	`0`	oui	Toujours présent avec de la valeur
`derived_exit_code_signal`	`0`	oui	Toujours présent avec de la valeur
`exit_code_status`	`0`	oui	Toujours présent avec de la valeur
`exit_code_signal`	`0`	oui	Toujours présent avec de la valeur
`node_details[0].name`	`"Hydra-MpiNG-abcdef01-2345-1"`	non	Toujours présent, mais `node_details` peut-être `"[]"`

Nom	Exemple de valeur	Obligatoire	Remarques
`node_details[0].instance_id`	`"i-0abcdef01234567a"`	non	Toujours présent, mais `node_details` peut-être `"[]"`
`node_details[0].instance_type`	`"t4g.micro"`	non	Toujours présent, mais `node_details` peut-être `"[]"`

Exemples de journaux d'achèvement des tâches

Les exemples suivants présentent les journaux d'achèvement des tâches pour différents types et états de tâches :


```
{ "jobcomp": { "job_id": 1, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T16:32:57",
 "end_time": "2025-06-19T16:33:03", "node_list": "Hydra-MpiNG-abcdef01-2345-
[1-2]", "node_cnt": 2, "proc_cnt": 2, "work_dir": "/usr/bin", "reservation_name":
 "", "tres": { "cpu": 2, "mem": { "val": 1944, "unit": "M" }, "node": 2,
 "billing": 2 }, "account": "", "qos": "", "wc_key": "", "cluster": "unknown",
 "submit_time": "2025-06-19T16:29:40", "eligible_time": "2025-06-19T16:29:41",
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc123def45678", "instance_type": "t4g.micro" }, { "name":
 "Hydra-MpiNG-abcdef01-2345-2", "instance_id": "i-0def456abc78901", "instance_type":
 "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 2, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T16:33:13",
 "end_time": "2025-06-19T16:33:14", "node_list": "Hydra-MpiNG-abcdef01-2345-
[1-2]", "node_cnt": 2, "proc_cnt": 2, "work_dir": "/usr/bin", "reservation_name":
 "", "tres": { "cpu": 2, "mem": { "val": 1944, "unit": "M" }, "node": 2,
 "billing": 2 }, "account": "", "qos": "", "wc_key": "", "cluster": "unknown",
 "submit_time": "2025-06-19T16:33:13", "eligible_time": "2025-06-19T16:33:13",
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
```

```
 "instance_id": "i-0abc123def45678", "instance_type": "t4g.micro" }, { "name":
 "Hydra-MpiNG-abcdef01-2345-2", "instance_id": "i-0def456abc78901", "instance_type":
 "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 3, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T22:58:57", "end_time":
 "2025-06-19T22:58:57", "node_list": "Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 972, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T22:55:46",
 "eligible_time": "2025-06-19T22:55:46", "derived_exit_code_status": 0,
 "derived_exit_code_signal": 0, "exit_code_status": 0, "exit_code_signal":
 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1", "instance_id":
 "i-0abc234def56789", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 4, "user": "root", "user_id": 0, "group": "root",
 "group_id": 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-
MpiQueue-abcdef01-7", "time_limit": "525600", "start_time": "2025-06-19T23:04:27",
 "end_time": "2025-06-19T23:04:27", "node_list": "Hydra-MpiNG-abcdef01-2345-
[1-2]", "node_cnt": 2, "proc_cnt": 2, "work_dir": "/root", "reservation_name":
 "", "tres": { "cpu": 2, "mem": { "val": 1944, "unit": "M" }, "node": 2,
 "billing": 2 }, "account": "", "qos": "", "wc_key": "", "cluster": "unknown",
 "submit_time": "2025-06-19T23:01:38", "eligible_time": "2025-06-19T23:01:38",
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc234def56789", "instance_type": "t4g.micro" }, { "name":
 "Hydra-MpiNG-abcdef01-2345-2", "instance_id": "i-0def345abc67890", "instance_type":
 "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 5, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "FAILED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:09:00", "end_time":
 "2025-06-19T23:09:00", "node_list": "(null)", "node_cnt": 0, "proc_cnt": 0,
 "work_dir": "/root", "reservation_name": "", "tres": { "cpu": 1, "mem": { "val":
 1, "unit": "G" }, "node": 1, "billing": 1 }, "account": "", "qos": "", "wc_key":
 "", "cluster": "unknown", "submit_time": "2025-06-19T23:09:00", "eligible_time":
 "2025-06-19T23:09:00", "derived_exit_code_status": 0, "derived_exit_code_signal": 0,
 "exit_code_status": 0, "exit_code_signal": 1, "node_details": [] } }
{ "jobcomp": { "job_id": 6, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "CANCELLED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:09:36",
 "end_time": "2025-06-19T23:09:36", "node_list": "(null)", "node_cnt": 0, "proc_cnt":
 0, "work_dir": "/root", "reservation_name": "", "tres": { "cpu": 1, "mem":
 { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "", "qos":
 "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:09:35",
 "eligible_time": "2025-06-19T23:09:36", "het_job_id": 6, "het_job_offset": 0,
```

```
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status": 0,
 "exit_code_signal": 1, "node_details": [] } }
{ "jobcomp": { "job_id": 7, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "CANCELLED", "partition": "Hydra-MpiQueue-
abcdef01-7", "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:10:03",
 "end_time": "2025-06-19T23:10:03", "node_list": "(null)", "node_cnt": 0, "proc_cnt":
 0, "work_dir": "/root", "reservation_name": "", "tres": { "cpu": 1, "mem":
 { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "", "qos":
 "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:10:03",
 "eligible_time": "2025-06-19T23:10:03", "het_job_id": 7, "het_job_offset": 0,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status": 0,
 "exit_code_signal": 1, "node_details": [] } }
{ "jobcomp": { "job_id": 8, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:11:24", "end_time":
 "2025-06-19T23:11:24", "node_list": "Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:11:23",
 "eligible_time": "2025-06-19T23:11:23", "het_job_id": 8, "het_job_offset": 0,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc234def56789", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 9, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:11:24", "end_time":
 "2025-06-19T23:11:24", "node_list": "Hydra-MpiNG-abcdef01-2345-2", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:11:23",
 "eligible_time": "2025-06-19T23:11:23", "het_job_id": 8, "het_job_offset": 1,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-2",
 "instance_id": "i-0def345abc67890", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 10, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:12:24", "end_time":
 "2025-06-19T23:12:24", "node_list":"Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 400, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:12:14",
 "eligible_time": "2025-06-19T23:12:14", "het_job_id": 10, "het_job_offset": 0,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
```

```
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc234def56789", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 11, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:12:24", "end_time":
 "2025-06-19T23:12:24", "node_list":"Hydra-MpiNG-abcdef01-2345-2", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 600, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:12:14",
 "eligible_time": "2025-06-19T23:12:14", "het_job_id": 10, "het_job_offset": 1,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-2",
 "instance_id": "i-0def345abc67890", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 13, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:47:57", "end_time":
 "2025-06-19T23:47:58", "node_list":"Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 972, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:43:56",
 "eligible_time": "2025-06-19T23:43:56" , "array_job_id": 12, "array_task_id": 1,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc345def67890", "instance_type": "t4g.micro" } ] } }
{ "jobcomp": { "job_id": 12, "user": "root", "user_id": 0, "group": "root", "group_id":
 0, "name": "wrap", "job_state": "COMPLETED", "partition": "Hydra-MpiQueue-abcdef01-7",
 "time_limit": "UNLIMITED", "start_time": "2025-06-19T23:47:58", "end_time":
 "2025-06-19T23:47:58", "node_list":"Hydra-MpiNG-abcdef01-2345-1", "node_cnt":
 1, "proc_cnt": 1, "work_dir": "/root", "reservation_name": "", "tres": { "cpu":
 1, "mem": { "val": 972, "unit": "M" }, "node": 1, "billing": 1 }, "account": "",
 "qos": "", "wc_key": "", "cluster": "unknown", "submit_time": "2025-06-19T23:43:56",
 "eligible_time": "2025-06-19T23:43:56" , "array_job_id": 12, "array_task_id": 2,
 "derived_exit_code_status": 0, "derived_exit_code_signal": 0, "exit_code_status":
 0, "exit_code_signal": 0, "node_details": [ { "name": "Hydra-MpiNG-abcdef01-2345-1",
 "instance_id": "i-0abc345def67890", "instance_type": "t4g.micro" } ] } }
```

Le planificateur se connecte à PCS AWS

Vous pouvez configurer AWS PCS pour envoyer des données de journalisation détaillées depuis votre planificateur de cluster à Amazon CloudWatch Logs, Amazon Simple Storage Service (Amazon S3) et Amazon Data Firehose. Cela peut faciliter la surveillance et le dépannage.

Table des matières

- Conditions préalables
- Configurer les journaux du planificateur
- Le planificateur enregistre les chemins et les noms des flux
- Exemple d'enregistrement du journal du planificateur

Conditions préalables

Le principal IAM qui gère le cluster AWS PCS doit autoriser l'`pcs:AllowVendedLogDeliveryForResource`action.

L'exemple de politique IAM suivant accorde les autorisations requises.

JSON

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

Configurer les journaux du planificateur

Vous pouvez configurer les journaux du planificateur pour votre cluster AWS PCS à l'aide du AWS Management Console ou. AWS CLI

AWS Management Console

Pour configurer les journaux du planificateur avec la console

1. Ouvrez la console AWS PCS.
2. Dans le panneau de navigation, choisissez Clusters.
3. Choisissez le cluster dans lequel vous souhaitez ajouter les journaux du planificateur.
4. Sur la page des détails du cluster, choisissez l'onglet Logs.
5. Sous Scheduler Logs, choisissez Ajouter pour ajouter jusqu'à 3 destinations de livraison de CloudWatch journaux parmi Logs, Amazon S3 et Firehose.
6. Choisissez Mettre à jour les livraisons de journaux.

AWS CLI

Pour configurer les journaux du planificateur à l'aide du AWS CLI

1. Créez une destination de livraison du journal :

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

Remplacez :

- *region*— L' Région AWS endroit où vous souhaitez créer la destination, tel que `us-east-1`
- *pcs-logs-destination*— Un nom pour la destination
- *resource-arn*— Le nom de ressource Amazon (ARN) d'un groupe de CloudWatch journaux Logs, d'un compartiment S3 ou d'un flux de diffusion Firehose.

Pour plus d'informations, consultez PutDeliveryDestinationle manuel Amazon CloudWatch Logs API Reference.

2. Définissez le cluster PCS comme source de livraison de journaux :

```
aws logs put-delivery-source --region region \
  --name cluster-logs-source-name \
```

```
  --resource-arn cluster-arn \
  --log-type PCS_SCHEDULER_LOGS
```

Remplacez :

- *region*— Le Région AWS nom de votre cluster, tel que `us-east-1`
- *cluster-logs-source-name*— Un nom pour la source
- *cluster-arn*— l'ARN de votre cluster AWS PCS

Pour plus d'informations, consultez PutDeliverySourcele manuel Amazon CloudWatch Logs API Reference.

3. Connectez la source de livraison à la destination de livraison :

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

Remplacez :

- *region*— Le Région AWS, tel que `us-east-1`
- *cluster-logs-source*— Le nom de votre source de livraison
- *destination-arn*— L'ARN de votre destination de livraison

Pour plus d'informations, consultez CreateDeliveryle manuel Amazon CloudWatch Logs API Reference.

Le planificateur enregistre les chemins et les noms des flux

Le chemin et le nom des journaux du planificateur AWS PCS dépendent du type de destination.

- CloudWatch Journaux
  - Un flux CloudWatch Logs suit cette convention de dénomination.

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

Example

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

- Compartiment S3
  - Le chemin de sortie d'un compartiment S3 suit cette convention de dénomination :

```
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.
```

  - Le nom d'un objet S3 suit cette convention :

```
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
```

Exemple d'enregistrement du journal du planificateur

AWS Les journaux du planificateur PCS sont structurés. Ils incluent des champs tels que l'identifiant du cluster, le type de planificateur, les versions majeures et de correctif, en plus du message de journal émis par le processus du contrôleur Slurm. Voici un exemple.

```
{
    "resource_id": "s3431v9rx2",
    "resource_type": "PCS_CLUSTER",
    "event_timestamp": 1721230979,
    "log_level": "info",
    "log_name": "slurmctld",
    "scheduler_type": "slurm",
    "scheduler_major_version": "25.05",
    "scheduler_patch_version": "3",
```

```
    "node_type": "controller_primary",
    "message": "[2024-07-17T15:42:58.614+00:00] Running as primary controller\n"
}
```

Surveillance d'un service de calcul AWS parallèle avec Amazon CloudWatch

Amazon CloudWatch assure le suivi de l'état et des performances de votre cluster AWS Parallel Computing Service (AWS PCS) en collectant des métriques à partir du cluster à intervalles réguliers. Ces indicateurs sont conservés, ce qui vous permet d'accéder aux données historiques et de mieux comprendre les performances de votre cluster au fil du temps.

CloudWatch vous permet également de surveiller les instances EC2 lancées par AWS PCS afin de répondre à vos exigences de scalabilité. Bien que vous puissiez inspecter les journaux des instances en cours d'exécution, CloudWatch les métriques et les données de journalisation sont généralement supprimées une fois les instances fermées. Cependant, vous pouvez configurer l' CloudWatch agent sur les instances à l'aide d'un modèle de lancement EC2 pour conserver les métriques et les journaux même après la fermeture de l'instance, ce qui permet une surveillance et une analyse à long terme.

Explorez les rubriques de cette section pour en savoir plus sur la surveillance de l'utilisation de AWS PC CloudWatch.

Rubriques

- Surveillance des métriques AWS PCS à l'aide CloudWatch
- Surveillance des instances AWS PCS à l'aide d'Amazon CloudWatch

Surveillance des métriques AWS PCS à l'aide CloudWatch

Vous pouvez surveiller l'état du cluster AWS PCS à l'aide d'Amazon CloudWatch, qui collecte les données de votre cluster et les transforme en métriques en temps quasi réel. Ces statistiques sont conservées pendant une période de 15 mois, afin que vous puissiez accéder aux informations historiques et avoir une meilleure idée des performances de votre cluster. Les métriques du cluster sont envoyées à des CloudWatch intervalles d'une minute. Pour plus d'informations CloudWatch, consultez Qu'est-ce qu'Amazon CloudWatch ? dans le guide de CloudWatch l'utilisateur Amazon.

AWS PCS publie les métriques suivantes dans l'espace de noms AWS/PCS dans. CloudWatch Ils ont une seule dimension,`ClusterId`.

Nom	Description	Uunités
ActualCapacity	IdleCapacity + UtilizedCapacity	Nombre
CapacityUtilization	UtilizedCapacity / ActualCapacity	Nombre
DesiredCapacity	ActualCapacity + PendingCapacity	Nombre
IdleCapacity	Nombre d'instances en cours d'exécution mais non allouées aux tâches	Nombre
UtilizedCapacity	Nombre d'instances en cours d'exécution et allouées aux tâches	Nombre

Surveillance des instances AWS PCS à l'aide d'Amazon CloudWatch

AWS PCS lance des instances Amazon EC2 selon les besoins pour répondre aux exigences de dimensionnement définies dans vos groupes de nœuds de calcul PCS. Vous pouvez surveiller ces instances lorsqu'elles sont en cours d'exécution à l'aide d'Amazon CloudWatch. Vous pouvez consulter les journaux des instances en cours d'exécution en vous y connectant et en utilisant des outils de ligne de commande interactifs. Toutefois, par défaut, CloudWatch les données des métriques ne sont conservées que pendant une période limitée une fois qu'une instance est résiliée, et les journaux d'instance sont généralement supprimés en même temps que les volumes EBS qui soutiennent l'instance. Pour conserver les métriques ou les données de journalisation des instances lancées par PCS après leur arrêt, vous pouvez configurer l' CloudWatch agent sur vos instances avec un modèle de lancement EC2. Cette rubrique fournit une vue d'ensemble de la surveillance des instances en cours d'exécution et fournit des exemples de configuration des métriques et des journaux d'instance persistants.

Surveillance des instances en cours d'exécution

Recherche d'instances AWS PCS

Pour surveiller les instances lancées par PCS, recherchez les instances en cours d'exécution associées à un cluster ou à un groupe de nœuds de calcul. Ensuite, dans la console EC2 d'une instance donnée, inspectez les sections État et alarmes et Surveillance. Si l'accès de connexion est configuré pour ces instances, vous pouvez vous y connecter et inspecter les différents fichiers journaux des instances. Pour plus d'informations sur l'identification des instances gérées par PCS, consultezRecherche d'instances de groupes de nœuds de calcul dans AWS PCS.

Permettre des métriques détaillées

Par défaut, les métriques d'instance sont collectées à intervalles de 5 minutes. Pour collecter des métriques à intervalles d'une minute, activez la CloudWatch surveillance détaillée dans votre modèle de lancement de groupe de nœuds de calcul. Pour de plus amples informations, veuillez consulter Activez la CloudWatch surveillance détaillée.

Configuration des métriques et des journaux d'instance persistants

Vous pouvez conserver les métriques et les journaux de vos instances en installant et en configurant l' CloudWatch agent Amazon sur celles-ci. Cela comprend trois étapes principales :

1. Créez une configuration d' CloudWatch agent.
2. Stockez la configuration dans un endroit où elle peut être récupérée par les instances PCS.
3. Rédigez un modèle de lancement EC2 qui installe le logiciel de l' CloudWatch agent, récupère votre configuration et démarre l' CloudWatch agent à l'aide de cette configuration.

Pour plus d'informations, consultez Collecter des métriques, des journaux et des traces avec l' CloudWatch agent dans le guide de CloudWatch l'utilisateur Amazon, etUtilisation des modèles de lancement Amazon EC2 avec PCS AWS.

Création d'une configuration CloudWatch d'agent

Avant de déployer l' CloudWatch agent sur vos instances, vous devez générer un fichier de configuration JSON qui spécifie les métriques, les journaux et les traces à collecter. Les fichiers de configuration peuvent être créés à l'aide d'un assistant ou manuellement à l'aide d'un éditeur de texte. Le fichier de configuration sera créé manuellement pour cette démonstration.

Sur un ordinateur sur lequel la CLI AWS est installée, créez un fichier CloudWatch de configuration nommé config.json avec le contenu ci-dessous. Vous pouvez également utiliser l'URL suivante pour télécharger une copie du fichier.

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

Remarques

- Les chemins de journal figurant dans le fichier d'exemple concernent Amazon Linux 2. Si vos instances doivent utiliser un système d'exploitation de base différent, modifiez les chemins comme il convient.
- Pour capturer d'autres journaux, ajoutez des entrées supplémentaires sous`collect_list`.
- Les valeurs saisies `{brackets}` sont des variables modélisées. Pour obtenir la liste complète des variables prises en charge, voir Création ou modification manuelle du fichier de configuration de l' CloudWatch agent dans le guide de CloudWatch l'utilisateur Amazon.
- Vous pouvez choisir d'omettre `logs` ou `metrics` de ne pas collecter ces types d'informations.

```
{
    "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
```

```
            }
        }
    }
}
```

Ce fichier demande à l' CloudWatch agent de surveiller plusieurs fichiers qui peuvent être utiles pour diagnostiquer les erreurs liées au démarrage des instances, à l'authentification et à la connexion, ainsi que dans d'autres domaines de résolution des problèmes. Il s'agit des licences suivantes :

- `/var/log/cloud-init.log`— Résultat de la phase initiale de configuration de l'instance
- `/var/log/cloud-init-output.log`— Résultat des commandes exécutées lors de la configuration de l'instance
- `/var/log/amazon/pcs/bootstrap.log`— Résultat des opérations spécifiques au PC exécutées lors de la configuration de l'instance
- `/var/log/slurmd.log`— Résultat du daemon slurmd du gestionnaire de charge de travail Slurm
- `/var/log/messages`— Messages système provenant du noyau, des services système et des applications
- `/var/log/secure`— Journaux liés aux tentatives d'authentification, tels que SSH, sudo et autres événements de sécurité

Les fichiers journaux sont envoyés à un groupe de CloudWatch journaux nommé`/PCSLogs/instances`. Les flux de journaux sont une combinaison de l'ID d'instance et du nom de base du fichier journal. Le groupe de journaux a une durée de conservation de 30 jours.

En outre, le fichier demande à l' CloudWatch agent de collecter plusieurs métriques communes, en les agrégeant par ID d'instance.

Stocker la configuration

Le fichier de configuration de l' CloudWatch agent doit être stocké de manière à être accessible aux instances du nœud de calcul PCS. Il existe deux méthodes courantes pour ce faire. Vous pouvez le télécharger dans un compartiment Amazon S3 auquel les instances de votre groupe de nœuds de calcul auront accès via leur profil d'instance. Vous pouvez également le stocker en tant que paramètre SSM dans Amazon Systems Manager Parameter Store.

Téléchargement vers un compartiment S3

Pour stocker votre fichier dans S3, utilisez les commandes de l'interface de ligne de commande AWS ci-dessous. Avant d'exécuter la commande, effectuez les remplacements suivants :

- *amzn-s3-demo-bucket*Remplacez-le par votre propre nom de compartiment S3

Tout d'abord (cela est facultatif si vous avez un bucket existant), créez un bucket pour contenir vos fichiers de configuration.

```
aws s3 mb s3://amzn-s3-demo-bucket
```

Ensuite, chargez le fichier dans le compartiment.

```
aws s3 cp ./config.json s3://amzn-s3-demo-bucket/
```

Stocker en tant que paramètre SSM

Pour enregistrer votre fichier en tant que paramètre SSM, utilisez la commande ci-dessous. Avant d'exécuter la commande, effectuez les remplacements suivants :

- Remplacez *region-code* par la région AWS dans laquelle vous travaillez avec AWS PCS.
- (Facultatif) *AmazonCloudWatch-PCS* Remplacez-le par votre propre nom pour le paramètre. Notez que si vous modifiez le préfixe du nom de, `AmazonCloudWatch-` vous devrez spécifiquement ajouter un accès en lecture au paramètre SSM dans le profil d'instance de votre groupe de nœuds.

```
aws ssm put-parameter \
   --region region-code \
   --name "AmazonCloudWatch-PCS" \
   --type String \
   --value file://config.json
```

Rédiger un modèle de lancement EC2

Les détails spécifiques du modèle de lancement varient selon que votre fichier de configuration est stocké dans S3 ou SSM.

Utiliser une configuration stockée dans S3

Ce script installe CloudWatch l'agent, importe un fichier de configuration depuis un compartiment S3 et lance l' CloudWatch agent avec celui-ci. Remplacez les valeurs suivantes dans ce script par vos propres informations :

- *amzn-s3-demo-bucket*— Le nom d'un compartiment S3 que votre compte peut lire
- */config.json*— Chemin relatif à la racine du compartiment S3 où la configuration est stockée

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

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

packages:
- amazon-cloudwatch-agent

runcmd:
- aws s3 cp s3://amzn-s3-demo-bucket/config.json /etc/s3-cw-config.json
- /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m
 ec2 -s -c file://etc/s3-cw-config.json

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

Le profil d'instance IAM du groupe de nœuds doit avoir accès au bucket. Voici un exemple de politique IAM pour le bucket dans le script de données utilisateur ci-dessus.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
```

```
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        }
    ]
}
```

Notez également que les instances doivent autoriser le trafic sortant vers le S3 et les points de CloudWatch terminaison. Cela peut être réalisé à l'aide de groupes de sécurité ou de points de terminaison VPC, en fonction de l'architecture de votre cluster.

Utiliser une configuration stockée dans SSM

Ce script installe CloudWatch l'agent, importe un fichier de configuration à partir d'un paramètre SSM et lance l' CloudWatch agent avec celui-ci. Remplacez les valeurs suivantes dans ce script par vos propres informations :

- (Facultatif) *AmazonCloudWatch-PCS* Remplacez-le par votre propre nom pour le paramètre.

```
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==--
```

La politique d'instance IAM pour le groupe de nœuds doit être CloudWatchAgentServerPolicyassociée à celle-ci.

Si le nom de votre paramètre ne commence pas par, `AmazonCloudWatch-` vous devrez spécifiquement ajouter un accès en lecture au paramètre SSM dans le profil d'instance de votre

groupe de nœuds. Voici un exemple de politique IAM qui illustre cela pour le préfixe*DOC-EXAMPLE-PREFIX*.

JSON

```
{
  "Version":"2012-10-17",
  "Statement" : [
    {
      "Sid" : "CustomCwSsmMParamReadOnly",
      "Effect" : "Allow",
      "Action" : [
        "ssm:GetParameter"
      ],
      "Resource" : "arn:aws:ssm:*:*:parameter/DOC-EXAMPLE-PREFIX*"
    }
  ]
}
```

Notez également que les instances doivent autoriser le trafic sortant vers le SSM et CloudWatch les points de terminaison. Cela peut être réalisé à l'aide de groupes de sécurité ou de points de terminaison VPC, en fonction de l'architecture de votre cluster.

Journalisation des appels d'API du service de calcul AWS parallèle à l'aide de AWS CloudTrail

AWS PCS est intégré à AWS CloudTrail un service qui fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un AWS service dans AWS PCS. CloudTrail capture tous les appels d'API pour AWS PCS sous forme d'événements. Les appels capturés incluent des appels provenant de la console AWS PCS et des appels de code vers les opérations de l'API AWS PCS. Si vous créez un suivi, vous pouvez activer la diffusion continue d' CloudTrail événements vers un compartiment Amazon S3, y compris des événements pour AWS PCS. Si vous ne configurez pas de suivi, vous pouvez toujours consulter les événements les plus récents dans la CloudTrail console dans Historique des événements. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été faite à AWS PCS, l'adresse IP à partir de laquelle la demande a été faite, qui a fait la demande, quand elle a été faite et des détails supplémentaires.

Pour en savoir plus CloudTrail, consultez le guide de AWS CloudTrail l'utilisateur.

AWS Informations PCS dans CloudTrail

CloudTrail est activé sur votre compte Compte AWS lorsque vous créez le compte. Lorsqu'une activité se produit dans AWS PCS, cette activité est enregistrée dans un CloudTrail événement avec d'autres événements de AWS service dans l'historique des événements. Vous pouvez consulter, rechercher et télécharger les événements récents dans votre Compte AWS. Pour plus d'informations, consultez la section Affichage des événements avec l'historique des CloudTrail événements.

Pour un enregistrement continu des événements de votre PC Compte AWS, y compris des événements pour AWS PCS, créez un parcours. Un suivi permet CloudTrail de fournir des fichiers journaux à un compartiment Amazon S3. Par défaut, lorsque vous créez un journal d'activité dans la console, il s'applique à toutes les régions Régions AWS. Le journal enregistre les événements de toutes les régions de la AWS partition et transmet les fichiers journaux au compartiment Amazon S3 que vous spécifiez. En outre, vous pouvez configurer d'autres AWS services pour analyser plus en détail les données d'événements collectées dans les CloudTrail journaux et agir en conséquence. Pour plus d'informations, consultez les ressources suivantes :

- Présentation de la création d'un journal de suivi
- CloudTrail services et intégrations pris en charge
- Configuration des notifications Amazon SNS pour CloudTrail
- Réception de fichiers CloudTrail journaux de plusieurs régions et réception de fichiers CloudTrail journaux de plusieurs comptes

Toutes les actions AWS PCS sont enregistrées CloudTrail et documentées dans le AWS Parallel Computing Service API Reference. Par exemple, les appels aux `CreateComputeNodeGroupUpdateQueue`, et `DeleteCluster` les actions génèrent des entrées dans les fichiers CloudTrail journaux.

Chaque événement ou entrée de journal contient des informations sur la personne ayant initié la demande. Les informations relatives à l'identité permettent de déterminer les éléments suivants :

- Si la demande a été faite avec les informations d'identification de l'utilisateur root ou Gestion des identités et des accès AWS (IAM).
- Si la demande a été effectuée avec les informations d'identification de sécurité temporaires d'un rôle ou d'un utilisateur fédéré.
- Si la demande a été faite par un autre AWS service.

Pour de plus amples informations, veuillez consulter l'élément userIdentity CloudTrail .

Comprendre les entrées des fichiers CloudTrail journaux à partir de AWS PCS

Un suivi est une configuration qui permet de transmettre des événements sous forme de fichiers journaux à un compartiment S3 que vous spécifiez. CloudTrail les fichiers journaux contiennent une ou plusieurs entrées de journal. Un événement représente une demande unique provenant de n'importe quelle source et inclut des informations sur l'action demandée, la date et l'heure de l'action, les paramètres de la demande, etc. CloudTrail les fichiers journaux ne constituent pas une trace ordonnée des appels d'API publics, ils n'apparaissent donc pas dans un ordre spécifique.

L'exemple suivant montre une entrée de CloudTrail journal pour une `CreateQueue` action.


```
{
    "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"
}
```

Points de terminaison et quotas de service pour PCS AWS

Les sections suivantes décrivent les points de terminaison et les quotas de service pour le AWS Parallel Computing Service (AWS PCS). Les quotas de service, anciennement appelés limites, sont le nombre maximum de ressources de service ou d'opérations pour votre Compte AWS.

Vous Compte AWS avez des quotas par défaut pour chaque AWS service. Sauf indication contraire, chaque quota est spécifique à une région. Vous pouvez demander des augmentations pour certains quotas, et d'autres quotas ne peuvent pas être augmentés.

Pour plus d'informations, veuillez consulter la rubrique Quotas du service AWS dans les Références générales AWS .

Table des matières

- Points de terminaison de service
- Quotas de service
  - Quotas internes
  - Quotas pertinents pour les autres AWS services

Points de terminaison de service

Nom de la région	Région	Point de terminaison	Protocole
USA Est (Ohio)	us-east-2	pcs.us-east-2.amazonaws.com pcs-fips.us-east-2.amazonaws.com pcs-fips.us-east-2.api.aws pcs.us-east-2.api.aws	HTTPS
USA Est (Virginie du Nord)	us-east-1	pcs.us-east-1.amazonaws.com	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
		pcs-fips.us-east-1.amazonaws.com pcs-fips.us-east-1.api.aws pcs.us-east-1.api.aws	
USA Ouest (Oregon)	us-west-2	pcs.us-west-2.amazonaws.com pcs-fips.us-west-2.amazonaws.com pcs-fips.us-west-2.api.aws pcs.us-west-2.api.aws	HTTPS
Asie-Pacifique (Singapour)	ap-southeast-1	pcs.ap-southeast-1.amazonaws.com pcs.ap-southeast-1.api.aws	HTTPS
Asie-Pacifique (Sydney)	ap-southeast-2	pcs.ap-southeast-2.amazonaws.com pcs.ap-southeast-2.api.aws	HTTPS
Asie-Pacifique (Tokyo)	ap-northeast-1	pcs.ap-northeast-1.amazonaws.com pcs.ap-northeast-1.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Europe (Francfort)	eu-central-1	pcs.eu-central-1.amazonaws.com pcs.eu-central-1.api.aws	HTTPS
Europe (Irlande)	eu-west-1	pcs.eu-west-1.amazonaws.com pcs.eu-west-1.api.aws	HTTPS
Europe (Londres)	eu-west-2	pcs.eu-west-2.amazonaws.com pcs.eu-west-2.api.aws	HTTPS
Europe (Stockholm)	eu-north-1	pcs.eu-north-1.amazonaws.com pcs.eu-north-1.api.aws	HTTPS
AWS GovCloud (USA Est)	us-gov-east-1	pièces. us-gov-east-1. amazonaws.com pcs-fips. us-gov-east-1. amazonaws.com pcs-fips. us-gov-east-1.api.aws pièces. us-gov-east-1.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
AWS GovCloud (US-Ouest)	us-gov-west-1	pièces. us-gov-west-1. amazonaws.com pcs-fips. us-gov-west-1. amazonaws.com pcs-fips. us-gov-west-1.api.aws pièces. us-gov-west-1.api.aws	HTTPS

Quotas de service

Nom	Par défaut	Ajustable	Description
Clusters	5	Oui	Le nombre maximum de clusters par Région AWS.

> **Note**
>
> Les valeurs par défaut sont les quotas initiaux définis par AWS. Ces valeurs par défaut sont distincts des valeurs réelles de quotas appliqués et des quotas de service maximaux possible. Pour plus d'informations, veuillez consulter la rubrique Terminologie des Service Quotas dans le Guide de l'utilisateur Service Quotas.

Ces quotas de service sont répertoriés sous AWS Parallel Computing Service (PCS) dans le AWS Management Console. Pour demander une augmentation de quota pour les valeurs indiquées comme ajustables, consultez la section Demander une augmentation de quota dans le Guide de l'utilisateur du Service Quotas.

> **Important**
>
> N'oubliez pas de vérifier le Région AWS réglage actuel dans le AWS Management Console.

Quotas internes

Les quotas suivants sont internes et non ajustables.

Nom	Par défaut	Ajustable	Description
Création simultanée de clusters	1	Non	Le nombre maximum de clusters dans l'`Creating`état par Région AWS.
Groupes de nœuds de calcul par cluster	10	Non	Nombre maximal de groupes de nœuds de calcul par cluster.
Files d'attente par cluster	10	Non	Le nombre maximum de files d'attente par cluster.

Quotas pertinents pour les autres AWS services

AWS PCS utilise d'autres AWS services. Vos quotas de service pour ces services ont un impact sur votre utilisation des AWS PCS.

Quotas de service Amazon EC2 ayant un impact sur le PCS AWS

- Demandes d'instance ponctuelles
- Exécution d'instances à la demande
- Modèles de lancement
- Versions du modèle de lancement
- Demandes d'API Amazon EC2

Pour plus d'informations, consultez les quotas de service Amazon EC2 dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Résolution des problèmes dans le service de calcul AWS parallèle

Les rubriques suivantes fournissent des conseils pour résoudre certains problèmes que vous pourriez rencontrer dans AWS PCS.

- mises à jour du cluster
- Problèmes de bootstrap du nœud de calcul
- Réglages personnalisés de Slurm
- Instances EC2 interrompues après le redémarrage
- Identité et accès
- Problèmes de redémarrage de Slurm

Une instance EC2 dans AWS PCS est arrêtée et remplacée après le redémarrage

Vue d'ensemble du problème

Après le redémarrage d'une instance EC2 d'un groupe de nœuds de calcul, AWS PCS met automatiquement fin à l'instance et la remplace.

Pourquoi cela se produit

AWS PCS ne prend pas en charge les redémarrages d'instances. Si une instance EC2 est redémarrée, AWS PCS considère que l'instance est défectueuse et la remplace. Si AWS PCS arrête et remplace continuellement vos instances, cela peut être dû au fait que quelque chose redémarre vos instances après leur lancement. Parmi les exemples, citons les redémarrages automatisés sur l'instance EC2 (par exemple, un redémarrage automatique après l'application de correctifs), l'automatisation externe à l'instance EC2 (telle qu'une application de gestion réseau), un autre AWS service (tel que AWS Systems Manager) ou un redémarrage manuel effectué par une personne.

Que faire

Vous pouvez consulter vos `slurmd` journaux `slurmctld` ou vos journaux pour voir si votre instance a été redémarrée. Pour plus d'informations, consultez Le planificateur se connecte à PCS AWS

et Surveillance des instances AWS PCS à l'aide d'Amazon CloudWatch. L'exemple d'entrée de `slurmctld` journal suivant indique que l'instance a redémarré :

Example

```
[2024-09-12T06:42:50.393+00:00] validate_node_specs: Node Login-1 unexpectedly rebooted
 boot_time=1726123354 last response=1726123285
```

Redémarrage à cause de l'application de correctifs

Un redémarrage est souvent nécessaire après l'application des correctifs. N'appliquez pas de correctifs directement à une instance EC2 faisant partie d'un groupe de nœuds de calcul AWS PCS. Si vous devez appliquer des correctifs à vos instances EC2, vous devez appliquer vos correctifs à une Amazon Machine Image (AMI) mise à jour et mettre à jour vos groupes de nœuds de calcul pour utiliser l'AMI mise à jour. Les nouvelles instances EC2 lancées par AWS PCS pour ces groupes de nœuds de calcul utiliseront l'AMI mise à jour (patchée). Pour de plus amples informations, veuillez consulter Images Amazon Machine personnalisées (AMIs) pour AWS PC.

Résoudre les problèmes d'amorçage et d'enregistrement des nœuds de calcul dans les PCS AWS

Lorsque les nœuds de calcul ne démarrent pas ou ne s'enregistrent pas correctement auprès de votre cluster AWS PCS, vous pouvez rencontrer les symptômes suivants :

- Les jobs ne démarrent pas
- Vous ne pouvez pas vous connecter aux instances dans AWS Systems Manager
- Les instances s'arrêtent de façon inattendue
- Les instances sont remplacées en permanence

Ces défaillances peuvent être causées par des problèmes lors du lancement de l'instance EC2 ou lors du processus d'amorçage du nœud de calcul AWS PCS. Cette rubrique décrit les procédures destinées à vous aider à résoudre les problèmes lors du processus de démarrage du nœud AWS PCS. Pour plus d'informations sur la résolution des problèmes de lancement d'instance EC2, consultez la section Résolution des problèmes de lancement d'instance Amazon EC2 dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Les échecs de Bootstrap se produisent lorsqu'une instance EC2 est lancée avec succès mais échoue pendant le processus d'adhésion au cluster AWS PCS. Le processus de bootstrap comprend deux phases principales :

- Enregistrement du nœud : l'instance EC2 appelle l'action API RegisterComputeNodeGroupInstance AWS PCS pour s'enregistrer auprès du service AWS PCS. Des défaillances peuvent survenir en raison des problèmes suivants :
  - Permissions
    - Mauvais profil d'instance
  - Réseaux
    - Impossible de se connecter aux points de terminaison AWS PCS
    - Point de terminaison AWS PCS mal configuré
    - Instance dans un sous-réseau public sans adresse IP publique
    - Instance multi-NIC dans un sous-réseau public
  - Secret du cluster
    - Le secret du cluster a été supprimé ou marqué pour suppression
- Intégration à Slurm — L'instance s'exécute `slurmd` et rejoint le cluster Slurm. Des défaillances peuvent survenir en raison des problèmes suivants :
  - Permissions
    - Configuration du groupe de sécurité
    - Slurmctld ne parvient pas à envoyer un ping au nœud de calcul
  - Configuration personnalisée de l'AMI
    - Pilotes NVIDIA manquants
    - ResumeTimeout atteint

Comment fonctionne Slurm sur PC AWS

Cela peut vous aider à comparer le fonctionnement standard de Slurm à celui de Slurm sur PC. AWS

Traitement standard des tâches avec Slurm

Les étapes suivantes se produisent dans le cadre du traitement standard des tâches Slurm :

1. Lorsque vous soumettez une tâche, `slurmctld` elle valide et la met en file d'attente.

2. Lorsque les ressources sont disponibles, `slurmctld` alloue les nœuds existants.
3. `slurmd`les démons exécutent des tâches sur les nœuds alloués.

Traitement des tâches Slurm sur PCS AWS

Les étapes suivantes se produisent dans le cadre du traitement des tâches AWS PCS :

1. Lorsque vous soumettez une tâche, `slurmctld` elle valide et la met en file d'attente.
2. Lorsqu'une capacité supplémentaire est nécessaire, AWS PCS utilise le modèle de lancement du groupe de nœuds de calcul pour lancer de nouvelles instances EC2.
3. Les nouvelles instances s'installent dans le cluster :
   a. Les instances s'enregistrent auprès de AWS PCS.
   b. Les instances rejoignent le cluster Slurm.
4. Lorsque les ressources sont prêtes, `slurmctld` alloue des nœuds (y compris ceux qui viennent d'être démarrés).
5. `slurmd`les démons exécutent des tâches sur les nœuds alloués.

Récupérez les journaux d'instance

La première étape pour résoudre les problèmes d'amorçage des nœuds de calcul consiste à récupérer les journaux de l'instance. Vous pouvez choisir l'une des méthodes suivantes :

AWS CLI

Récupérez la sortie de console depuis le nœud de calcul à l'aide de la commande suivante :

```
aws ec2 get-console-output --region us-east-1 --instance-id i-1234567890abcdef0 --
output text
```

Remplacez *us-east-1* par votre AWS région et *i-1234567890abcdef0* par l'ID de votre instance.

AWS Systems Manager

Si vous pouvez vous connecter à l'instance à l'aide de Systems Manager, vous pouvez consulter directement le fichier journal du bootstrap :

1. Connectez-vous à l'instance à l'aide de Systems Manager. Pour plus d'informations, reportez-vous à la section Démarrage d'une session dans le Guide de l'utilisateur de Systems Manager.
2. Consultez le fichier journal du bootstrap :

```
sudo cat /var/log/amazon/pcs/bootstrap.log
```

> **Note**
>
> En cas de problème pendant la phase d'initialisation, vous devrez peut-être attendre environ 20 minutes avant de pouvoir vous connecter à l'instance. Systems Manager et les services SSH ne démarrent qu'une fois l'initialisation terminée ou lorsque l'exécution du bootstrap atteint un délai d'expiration en cas d'échec.

Récupérer VPC/Subnet/Security des groupes à partir d'un ID d'instance

Pour résoudre les problèmes liés à vos nœuds de calcul, vous devrez peut-être récupérer des informations sur le VPC, le sous-réseau et les groupes de sécurité associés à vos instances. Si vous ne connaissez pas votre instance IDs, consultezRecherche d'instances de groupes de nœuds de calcul dans AWS PCS.

AWS Management Console

Pour obtenir un VPC, un sous-réseau et des groupes de sécurité

1. Ouvrez la console Amazon EC2.
2. Choisissez Instances.
3. Dans le tableau Instances, choisissez l'ID de l'instance.
4. Trouvez l'ID VPC et l'ID de sous-réseau dans le résumé de l'instance affiché pour l'instance.
5. Dans le résumé de l'instance, choisissez l'onglet Sécurité.
6. Trouvez les groupes de sécurité dans l'onglet Sécurité.

AWS CLI

Utilisez la commande suivante pour récupérer les informations relatives au VPC, au sous-réseau et au groupe de sécurité pour votre instance :

```
aws ec2 describe-instances --instance-ids i-1234567890abcdef0 --query
 'Reservations[*].Instances[*].
{InstanceId:InstanceId,VpcId:VpcId,SubnetId:SubnetId,SecurityGroups:SecurityGroups[*].GroupI
 --output table
```

Problèmes d'enregistrement des nœuds

L'enregistrement d'un nœud est la première action exécutée par un nœud de calcul pendant le bootstrap. Le nœud appelle le point de terminaison de l'API AWS PCS pour s'enregistrer auprès du AWS PCS. Les échecs d'enregistrement affichent généralement des messages d'erreur similaires aux suivants :

```
<13>Nov  5 08:10:27 user-data: Recipe: aws-pcs-environment::node_registration
<13>Nov  5 08:10:27 user-data:   * ruby_block[Register NodeGroup Instance] action
 run[2024-11-05T08:10:27+00:00] INFO: Processing ruby_block[Register NodeGroup
 Instance] action run (aws-pcs-environment::node_registration line 19)
<13>Nov  5 08:15:46 user-data:
<13>Nov  5 08:15:46 user-data:
<13>Nov  5 08:15:46 user-data:
 ================================================================================
<13>Nov  5 08:15:46 user-data:     Error executing action `run` on resource
 'ruby_block[Register NodeGroup Instance]'
<13>Nov  5 08:15:46 user-data:
 ================================================================================
<13>Nov  5 08:15:46 user-data:
<13>Nov  5 08:15:46 user-data:     EOFError
```

Mauvais profil d'instance

Si l'instance ne parvient pas à s'enregistrer, vérifiez que le profil d'instance associé au nœud de calcul dispose de l'`pcs:RegisterComputeNodeGroupInstance`autorisation.

Pour plus d'informations sur la création d'un profil d'instance valide, consultezCréation d'un profil d'instance pour AWS PCS.

Impossible de se connecter aux points de terminaison AWS PCS

Si vos nœuds de calcul se trouvent dans un sous-réseau privé, assurez-vous que vous avez configuré des points de terminaison VPC AWS pour les PC ou que votre sous-réseau dispose d'une

route vers une passerelle NAT pour accéder à Internet. Pour plus d'informations, consultez les ressources suivantes :

- Accédez à un AWS service à l'aide d'un point de terminaison VPC d'interface dans le guide Amazon Virtual Private Cloud AWS PrivateLink.
- Points de terminaison et quotas de service pour PCS AWS.
- Connectez votre VPC à d'autres réseaux dans le guide de l'utilisateur Amazon Virtual Private Cloud
- AWS Mise en réseau PCS

Point de terminaison AWS PCS mal configuré

Si un message d'erreur similaire au suivant s'affiche, vérifiez la politique associée à votre point de terminaison AWS VPC PCS :

```
com.amazon.coral.security.AccessDeniedException: User: arn:aws:sts::xxx:assumed-
role/rolename/i-instanceid is not authorized to perform:
 pcs:RegisterComputeNodeGroupInstance on resource: arn:aws:pcs:us-west-2:xxx:cluster/
cluster-id as either the resource does not exist, some policy explicitly denies access,
 or no policy grants access
```

Pour plus d'informations sur la configuration des points de terminaison d'interface VPC pour AWS PCS, consultez. Accès AWS Parallel Computing Service via un point de terminaison d'interface (AWS PrivateLink)

Instance dans un sous-réseau public sans adresse IP publique

Si l'attribution automatique d'adresses IP publiques n'est pas activée dans votre sous-réseau et que la configuration de votre route utilise une passerelle Internet, les instances ne peuvent pas communiquer avec l'API AWS PCS.

Les instances d'un sous-réseau doté d'une passerelle Internet doivent avoir une adresse IP publique. Pour résoudre ce problème, choisissez l'une des options suivantes :

- Ajoutez un point de terminaison VPC pour AWS PCS à votre VPC de cluster. Cela permet aux instances de communiquer avec les AWS PCS sans qu'il soit nécessaire qu'une adresse IP publique passe par la passerelle Internet.
- Utilisez un sous-réseau privé avec une passerelle NAT, de sorte qu'aucune adresse IP publique ne soit requise.

- Activez l'attribution automatique d'adresses IP publiques via votre sous-réseau ou votre modèle de lancement afin que les instances puissent contacter l'API via la passerelle Internet. Notez que cette option n'est pas valide pour les instances d'interface multi-réseaux.

Instance multi-NIC dans un sous-réseau public

Vous devez utiliser un sous-réseau privé si vous utilisez un type d'instance doté de plusieurs interfaces réseau (NICs).

AWS les adresses IP publiques ne peuvent être attribuées qu'aux instances lancées avec une seule interface réseau. Pour plus d'informations sur les adresses IP, consultez la section Attribuer une IPv4 adresse publique lors du lancement de l'instance dans le Guide de l'utilisateur Amazon EC2 pour les instances Linux.

Les types d'instances multi-NIC nécessitent une passerelle NAT ou un proxy interne dans le sous-réseau pour accéder au point de terminaison AWS PCS. Vous pouvez également ajouter un point de terminaison VPC pour AWS PCS à votre VPC de cluster.

Le secret du cluster a été supprimé ou marqué pour suppression

Si le secret partagé Slurm dans AWS Secrets Manager a été supprimé ou marqué pour suppression, les nœuds de calcul ne seront pas enregistrés et votre cluster sera altéré.

AWS PCS crée automatiquement un secret partagé Slurm dans AWS Secrets Manager (avec le format de nom :`pcs!slurm-secret-<cluster-id>`) lorsque vous créez un cluster. Ce secret est requis pour sécuriser les communications dans le cluster. Pour de plus amples informations, veuillez consulter Utilisation des secrets de cluster dans AWS PCS.

Si ce secret est supprimé ou marqué pour suppression, les nouveaux nœuds ne pourront pas rejoindre le cluster et le contrôleur ou les autres démons du cluster (tels que `slurmd` et`slurmdbd`) ne pourront peut-être pas rejoindre le cluster en cas de redémarrage.

Pour résoudre ce problème, vous pouvez restaurer le secret supprimé s'il se trouve toujours dans la fenêtre de restauration. Pour des instructions détaillées, voir Restaurer un secret du Gestionnaire de AWS Secrets.

Si la fenêtre de restauration expire, le secret ne peut pas être restauré et le cluster AWS PCS concerné ne peut pas être restauré. Vous devez créer un nouveau cluster avec la même configuration. AWS PCS crée automatiquement un nouveau secret du planificateur.

Problèmes de jointure au cluster Slurm

Une fois l'enregistrement du nœud réussi, le nœud de calcul tente de rejoindre le cluster Slurm. Le `slurmd` démon du nœud contacte le contrôleur Slurm pour s'enregistrer auprès du cluster. Les échecs de jointure Slurm affichent généralement des messages d'erreur similaires aux suivants :

```
<13>Nov  5 17:20:29 user-data: [2024-11-05T17:20:28+00:00] FATAL:
 Mixlib::ShellOut::ShellCommandFailed: service[slurmd] (aws-pcs-slurm::finalize_slurm
 line 18) had an error: Mixlib::ShellOut::ShellCommandFailed: Expected process to exit
 with [0], but received '1'
<13>Nov  5 17:20:29 user-data: ---- Begin output of ["/usr/bin/systemctl", "--system",
 "start", "slurmd"] ----
<13>Nov  5 17:20:29 user-data: STDOUT:
<13>Nov  5 17:20:29 user-data: STDERR: Job for slurmd.service failed because the
 control process exited with error code. See "systemctl status slurmd.service" and
 "journalctl -xe" for details.
<13>Nov  5 17:20:29 user-data: ---- End output of ["/usr/bin/systemctl", "--system",
 "start", "slurmd"] ----
```

Configuration du groupe de sécurité

Vérifiez que vos groupes de sécurité sont correctement configurés pour permettre la communication entre les nœuds de calcul et le contrôleur Slurm. Les groupes de sécurité doivent autoriser le trafic suivant :

- Port 6817 pour `slurmd` communiquer avec `slurmctld`
- Port 6818 pour envoyer un `slurmctld` ping `slurmd`

Pour plus d'informations sur les exigences relatives aux groupes de sécurité, consultez les rubriques suivantes :

- Création de groupes de sécurité pour AWS PCS
- Création de modèles de lancement pour AWS PCS
- Exigences et considérations relatives aux groupes de sécurité

> **Important**
>
> Le groupe de sécurité du cluster que vous avez associé à votre cluster lors de la création du cluster doit également être configuré dans les groupes de sécurité de votre groupe de nœuds de calcul pour permettre aux nœuds de calcul de communiquer avec le contrôleur.

Pilotes NVIDIA manquants

Si l'instance démarre correctement mais que les tâches ne démarrent pas et que des messages d'erreur similaires aux suivants s'affichent dans les journaux de votre instance, il se peut que des pilotes NVIDIA soient manquants :

```
<13>Dec  2 13:52:00 user-data: [2024-12-02T13:52:00.094+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_config_always.sh: INFO: nvidia-smi not found!
...
<13>Dec  2 13:54:10 user-data: Job for slurmd.service failed because the control
 process exited with error code. See "systemctl status slurmd.service" and "journalctl
 -xe" for details.
<13>Dec  2 13:54:12 user-data: [2024-12-02T13:54:12.718+00:00] - /opt/aws/pcs/bin/
pcs_bootstrap_finalize.sh: INFO: systemctl could not start slurmd!
```

Si vous vous connectez à l'instance et vérifiez l'état du `slurmd` démon, une erreur similaire à la suivante peut s'afficher :

```
$ systemctl status slurmd
...
fatal: can't stat gres.conf file /dev/nvidia0: No such file or directory
```

Pour résoudre ce problème, installez les pilotes NVIDIA sur votre AMI personnalisée. Pour de plus amples informations, veuillez consulter Étape 4 — (Facultatif) Installation de pilotes, de bibliothèques et de logiciels d'application supplémentaires.

ResumeTimeout atteint

Si un nœud de calcul et son instance EC2 sont interrompus en raison d'un dysfonctionnement du nœud, il est possible que le AWS PCS ne prenne pas en charge l'AMI ou qu'il y ait des problèmes de réseau. L'instance EC2 s'exécute pendant environ 30 minutes jusqu'à ce que Slurm ResumeTimeout soit atteint et marque le nœud comme. `DOWN`

Si l'instance ne démarre pas correctement et n'est pas enregistrée auprès de AWS PCS (aucun `RegisterComputeNodeGroupInstance` appel pour l'instance EC2), consultez les journaux de votre instance pour détecter les messages d'erreur similaires aux suivants :

```
/opt/aws/pcs/bin/pcs_bootstrap_init.sh: No such file or directory
```

Cette erreur indique que le logiciel de démarrage du AWS PCS ne fait pas partie de l'AMI. Pour résoudre ce problème, assurez-vous que votre AMI personnalisée inclut le logiciel de démarrage AWS PCS. Pour de plus amples informations, veuillez consulter Images Amazon Machine personnalisées (AMIs) pour AWS PC.

Slurmctld ne parvient pas à envoyer un ping au nœud de calcul

Si l'instance exécute correctement la procédure d'amorçage et qu'elle est enregistrée auprès de AWS PCS, mais qu'elle n'`slurmctld`est pas en mesure de la voir et de lui soumettre des tâches, l'instance est définie sur `DOWN` après un certain temps, puis résiliée.

Cela peut être dû à des groupes de sécurité mal configurés. Par exemple, si le port 6817 est activé pour permettre de `slurmd` communiquer avec`slurmctld`, mais que le port 6818 est absent `slurmctld` pour autoriser le ping. `slurmd`

Vérifiez que vos groupes de sécurité incluent toutes les règles requises, comme indiqué dansExigences et considérations relatives aux groupes de sécurité.

Historique du document pour le guide de l'utilisateur du AWS PCS

Le tableau suivant décrit les modifications importantes apportées à la documentation de AWS PCS.

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
10 mars 2026	Agent PCS mis à jour	Mise à jour de la rubrique AMI pour l'agent AWS PCS 1.3.2-1. Correction d'un problème affectant le bootstrap des nœuds de calcul RHEL 8.10 et Rocky Linux 8.10. Pour plus d'informations, consultez Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS et AWS Versions de l'agent PCS.	N/A
11 février 2026	AWS PCS sorti en Asie-Pacifique (Mumbai) et en Europe (Paris)	AWS Le PCS est désormais disponible en Asie-Pacifique (Mumbai) (ap-south-1) et en Europe (Paris) (eu-west-3). CloudFormation des modèles sont disponibles pour démarrer en Asie-Pacifique (Mumbai) Région AWS et en Europe (Paris) Région AWS. Pour plus d'informations, consultez	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
		CloudFormation À utiliser pour créer un exemple de cluster AWS PCS et CloudFormation modèles pour créer un exemple de cluster AWS PCS.	
18 novembre 2025	Nouvelle fonctionnalité : API REST de Slurm	L'API REST de Slurm est désormais prise en charge pour Slurm 25.05 ou version ultérieure. Pour de plus amples informations, veuillez consulter API REST Slurm sur PCS AWS.	SDK AWS : 18/11/2020

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
10 novembre 2025	Nouvelle fonctionnalité : prise en charge du plugin de filtre Slurm CLI	AWS PCS prend désormais en charge les plugins de filtre Slurm CLI pour exécuter des scripts Lua personnalisés qui valident et modifient les paramètres de soumission des tâches avant qu'elles n'atteignent le contrôleur Slurm. Utilisez les filtres CLI pour appliquer des politiques personnalisées, définir des paramètres par défaut et fournir des conseils aux utilisateurs lors de la soumission des tâches. Cette fonctionnalité nécessite la version 25.05 ou ultérieure de Slurm. Pour de plus amples informations, veuillez consulter Utilisez les plugins de filtre Slurm CLI pour personnaliser la soumission des tâches dans PCS AWS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
7 novembre 2025	Agent PCS mis à jour	Mise à jour de la rubrique AMI pour l'agent AWS PCS 1.3.1-1. Pour plus d'informations, consultez Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS et AWS Versions de l'agent PCS.	N/A
3 novembre 2025	Agent PCS et installateurs Slurm mis à jour	Mise à jour de la rubrique AMI pour l'agent AWS PCS 1.3.0-1 et les programmes d'installation de Slurm 24.11.6-2, 24.05.8-2 et 23.11.10-4. Liste mise à jour des systèmes d'exploitation pris en charge. Pour plus d'informations, consultez Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS et AWS Versions de l'agent PCS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
23 octobre 2025	Contenu mis à jour : pcs-multi-cluster-login - configure.sh	Correction de certaines erreurs dans le script de configuration du nœud de connexion multi-clusters. Pour de plus amples informations, veuillez consulter AWS Code de script de configuration du nœud de connexion multi-cluster PCS.	N/A
21 octobre 2025	Nouvelle fonctionnalité : rotation secrète du cluster	AWS Le PCS prend désormais en charge la rotation des secrets du cluster pour améliorer la sécurité. Pour de plus amples informations, veuillez consulter Rotation des secrets de cluster dans AWS PCS. Autorisations d'administrateur minimales mises à jour pour prendre en charge la rotation des secrets du cluster. Pour de plus amples informations, veuillez consulter Autorisations minimales pour les AWS PCS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
17 octobre 2025	Nouveau sujet : script de configuration du nœud de connexion multi-clusters	Ajout d'une nouvelle rubrique qui fournit un script permettant de configurer un nœud de connexion autonome pour se connecter à plusieurs clusters AWS PCS. Le script automatise la configuration de plusieurs `sackd` démons Slurm et crée des scripts d'activation pour l'interaction avec les clusters. Pour de plus amples informations, veuillez consulter Connexion d'un nœud de connexion autonome à plusieurs clusters dans AWS PCS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
16 octobre 2025	Mise à jour pour Slurm 25.05	Mise à jour du guide de l'utilisateur pour le support de Slurm 25.05. Slurm 25.05 est désormais la version par défaut. Pour plus d'informations, consultez les ressources suivantes : • Versions Slurm en PCS AWS • Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS • Notes de mise à jour pour un exemple de AWS PCS AMIs	N/A
16 octobre 2025	Agent PCS mis à jour	Mise à jour de la rubrique AMI pour l'agent AWS PCS 1.2.2-1. Pour plus d'informations, consultez Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS et AWS Versions de l'agent PCS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
2 octobre 2025	Nouvelles fonctionn alités : redémarrage du nœud Slurm, mises à jour du cluster et paramètres personnalisés de Slurm	AWS PCS ajoute la prise en charge de nombreuse s nouvelles fonctionn alités : • Redémarrage du nœud Slurm : utilisez la `scontrol reboot` commande native de Slurm pour redémarre r les nœuds de calcul sans remplacer l'instance. Pour de plus amples informati ons, veuillez consulter Redémarrage de nœuds de calcul avec Slurm sur PCS AWS. • Mises à jour du cluster : modifiez les configura tions du cluster après la création sans avoir à le reconstruire. Pour de plus amples informati ons, veuillez consulter Mettre à jour un cluster dans AWS PCS. • Paramètres personnal isés de Slurm : configurez les paramètres avancés de Slurm sur les ressource s du cluster, de la file	01/10/2025

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
		d'attente et du groupe de nœuds de calcul. Pour de plus amples informations, veuillez consulter Configuration des paramètres personnalisés de Slurm dans PCS AWS.	
23 septembre 2025	Nouvelle rubrique de résolution des problèmes : Problèmes d'amorçage du nœud de calcul	Ajout de conseils de dépannage pour diagnostiquer et résoudre les problèmes de démarrage des nœuds de calcul. Pour de plus amples informations, veuillez consulter Résoudre les problèmes d'amorçage et d'enregistrement des nœuds de calcul dans les PCS AWS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
17 septembre	Nouvelle fonctionnalité : Capacity Blocks for ML	AWS PCS prend désormais en charge les blocs de capacité Amazon EC2 pour le ML, qui vous permettent de réserver des instances de calcul accéléré basées sur le GPU pour vos clusters. Pour de plus amples informations, veuillez consulter Utilisation des blocs de capacité Amazon EC2 pour le ML avec PCS AWS. Les autorisations minimales pour prendre en charge les blocs de capacité font désormais partie des autorisations minimales d'un administrateur de service. Pour de plus amples informations, veuillez consulter Autorisations minimales pour les AWS PCS.	17/09/2025

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
11 septembre 2025	Mise à jour des politiques gérées par AWS	AWS PCS les a mis à jour AWSPCSService RolePolicy pour prendre en charge les blocs de capacité. Pour de plus amples informations, veuillez consulter AWS politiques gérées pour le service de calcul AWS parallèle.	N/A
14 août 2025	Documentation sur le profil d'instance mise à jour	La documentation sur les profils d'instance a été améliorée avec des instructions CLI complètes pour créer des rôles IAM et des profils d'instance. step-by-stepDes procédures ont été ajoutées pour configurer des profils d'instance à l'aide du PCS AWS CLI et des instructions améliorées pour trouver les profils d'instance utilisés avec AWS PCS. Pour de plus amples informations, veuillez consulter Profils d'instance IAM pour AWS Parallel Computing Service.	14/08/2025

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
1er août 202	Nouveau sujet : plugins SPANK	Ajout de la documentation pour les plugins SPANK (Slurm Plug-in Architecture for Node and job Kontrol) que vous pouvez utiliser pour étendre et modifier le comportement de Slurm lors du lancement et de l'exécution des tâches sur des clusters PCS. AWS Pour plus d’informations, consultez Étendez les fonctionnalités de Slurm sur AWS PC avec les plugins SPANK.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
1er août 202	IPv6 support réseau	Ajout du support pour la IPv6 mise en réseau lors de la création de clusters AWS PCS. Vous pouvez désormais choisir IPv6 le type de réseau pour votre cluster, avec les mises à jour correspondantes des exigences VPC, de la configuration des sous-réseaux, des paramètre s du groupe de sécurité et des procédures de création de cluster. Pour plus d’informa tions, consultez AWS Exigences et considéra tions relatives au PCS, au VPC et aux sous-réseaux et Création d'un cluster dans AWS PCS.	2025-08-01

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
3 juillet 2025	AWS PCS sorti en Europe (Londres)	AWS Le PCS est désormais disponible en Europe (Londres) (eu-west-2). CloudFormation des modèles sont disponibles pour démarrer en Europe (Londres) Région AWS. Pour plus d'informations, consultez CloudFormation À utiliser pour créer un exemple de cluster AWS PCS et CloudFormation modèles pour créer un exemple de cluster AWS PCS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
1er juillet 2025	Instructions de console mises à jour	Vous pouvez désormais demander à AWS PCS de créer un profil d'instance de base et un groupe de sécurité pour vous lorsque vous créez un cluster et un groupe de nœuds de calcul dans la console. Pour en savoir plus, consultez : • Création d'un cluster dans AWS PCS • Création d'un groupe de nœuds de calcul dans AWS PCS • Profils d'instance IAM pour AWS Parallel Computing Service	N/A
23 juin 2025	Nouvelle politique gérée : AWSPCSCompute NodePolicy	Ajout d'une nouvelle politique gérée qui autorise les nœuds de calcul AWS PCS à se connecter aux clusters AWS PCS. Pour plus d'informations, consultez AWS politique gérée : AWSPCSCompute NodePolicy.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
19 juin 2025	Nouveau sujet : journaux d'achèvement des tâches	Utilisez les journaux d'achèvement des tâches pour enregistrer les détails des tâches une fois celles-ci terminées , sans frais supplémen taires. Pour de plus amples informations, veuillez consulter Journaux d'achèvement des tâches dans AWS PCS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
18 juin 2025	AWS Sortie du PCS en AWS GovCloud (US)	AWS Le PCS est désormais disponible en AWS GovCloud (US-Est) (us-gov-east-1) et AWS GovCloud (US-Ouest) (us-gov-west-1). CloudFormation des modèles sont disponibles pour démarrer dans le AWS GovCloud (US) Regions. Pour plus d'informations, consultez CloudFormation À utiliser pour créer un exemple de cluster AWS PCS et CloudFormation modèles pour créer un exemple de cluster AWS PCS. Pour plus d'informations sur les points de terminaison du service AWS PCS dans AWS GovCloud (US) Regions, consultezPoints de terminaison et quotas de service pour PCS AWS. Pour plus d'informations sur les différences entre AWS GovCloud (US) Regions, voir AWS PCS in AWS GovCloud (US)	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
		dans le guide de AWS GovCloud (US) l'utilisa teur.	
18 juin 2025	Agent PCS mis à jour	Mise à jour de la rubrique AMI pour l'agent AWS PCS 1.2.1-1. Pour de plus amples informations, veuillez consulter Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS.	N/A
15 mai 2025	Nouvelle fonctionnalité : comptabilité	La comptabilité Slurm est désormais prise en charge pour Slurm 24.11 ou version ultérieure. Pour de plus amples informations, veuillez consulter Comptabilité Slurm dans PCS AWS.	KIT DE DÉVELOPPE MENT LOGICIEL AWS : 15/05/2020

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
15 mai 2025	Mise à jour pour Slurm 24.11	Mise à jour du guide de l'utilisateur pour le support de Slurm 24.11.5. Pour plus d'informations, consultez les ressources suivantes : • Versions Slurm en PCS AWS • Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS • Notes de mise à jour pour un exemple de AWS PCS AMIs	N/A
5 mai 2025	FAQ sur les versions mises à jour de Slurm	Mise à jour des questions fréquemment posées (FAQ) sur les versions de Slurm sur les versions de Slurm en fin de vie (EOL) ou après cette date. Pour de plus amples informations, veuillez consulter Questions fréquemment posées sur les versions de Slurm dans PCS AWS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
17 avril 2025	Nouveau sujet : comment obtenir les détails d'un groupe de nœuds de calcul	Découvrez comment obtenir des informations sur un groupe de nœuds de calcul AWS PCS, telles que son ID, son ARN et son ID AMI. Pour de plus amples informations, veuillez consulter Obtenez les détails du groupe de nœuds de calcul dans AWS PCS.	N/A
2 avril 2025	Programme d'installation de Slurm mis à jour	Mise à jour de la rubrique AMI pour le programme d'installation de Slurm 24.05.7-1. Pour de plus amples informations, veuillez consulter Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS.	N/A
28 mars 202	Quotas ajoutés pour le nombre maximal de groupes de nœuds de calcul et de files d'attente	Ajout de quotas internes non ajustables pour le nombre maximum de groupes de nœuds de calcul par cluster et le nombre maximum de files d'attente par cluster. Pour de plus amples informations, veuillez consulter Quotas internes.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
14 mars 2025	Modification d'une clé de propriété dans le CloudFormation modèle	`Id`est maintenant `TemplateId` pour la `CustomLaunchTemplate` propriété dans le CloudFormation modèle. Pour plus d'informations, consultez Ressources dans Éléments d'un CloudFormation modèle pour AWS PCS.	N/A
13 mars 202	Informations de version ajoutées pour l'agent AWS PCS et Slurm	Ajout d'une nouvelle rubrique qui décrit les modifications apportées à chaque version de l'agent AWS PCS. Pour de plus amples informations, veuillez consulter AWS Versions de l'agent PCS. Ajout d'informations supplémentaires à la rubrique sur les versions de Slurm qui décrit les dates de support importantes et les notes de version détaillées concernant le support AWS PCS pour Slurm. Pour de plus amples informations, veuillez consulter Versions Slurm en PCS AWS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
07 mars 202	Agent PCS mis à jour	Mise à jour de la rubrique AMI pour l'agent AWS PCS 1.2.0-1. Pour de plus amples informations, veuillez consulter Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS.	N/A
03 février 20	Ajout d'un sujet sur l'utilisation AWS CloudFormation avec AWS PCS	Ajout d'une rubrique au guide de l'utilisateur qui fournit un exemple d'utilisation CloudFormation avec AWS PCS. Cette rubrique fournit une procédure permettant d'utiliser un exemple de CloudFormation modèle pour créer l'exemple de cluster AWS PCS, et décrit brièvement les sections de ce modèle. Pour de plus amples informations, veuillez consulter Commencez avec CloudFormation AWS PCS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
18 décembre	Mise à jour pour Slurm 24.05	Mise à jour du guide de l'utilisateur pour le support de Slurm 24.05. Pour plus d'informations, consultez Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS et Notes de mise à jour pour un exemple de AWS PCS AMIs.	N/A
18 décembre	Exemple de versions NVIDIA mises à jour pour Slurm 23.11 AMIs	Version du pilote NVIDIA et de CUDA mise à jour dans l'exemple Slurm 23.11. AMIs Pour de plus amples informations, veuillez consulter Notes de mise à jour pour un exemple de AWS PCS AMIs.	N/A
17 décembre	Programme d'installation de Slurm mis à jour	Mise à jour de la rubrique AMI pour le programme d'installation de Slurm 23.11.10-3. Pour de plus amples informations, veuillez consulter Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
13 décembre	Agent PCS mis à jour	Mise à jour de la rubrique AMI pour l'agent AWS PCS 1.1.1-1. Pour de plus amples informations, veuillez consulter Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS.	N/A
6 décembre 2	Agent PCS et programme d'installation de Slurm mis à jour	Mise à jour de la rubrique AMI pour l'agent AWS PCS 1.1.0-1 et le programme d'installation de Slurm 23.11.10-2. Pour de plus amples informations, veuillez consulter Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS.	N/A
6 décembre 2	Ajout d'une rubrique sur le support du système d'exploitation	Pour de plus amples informations, veuillez consulter Systèmes d'exploitation pris en charge sur AWS PCS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
8 novembre 2	Guide de l'utilisateur réorganisé	Nous avons réorganisé le guide de l'utilisateur pour placer les sujets au niveau supérieur, déplacé certains sujets vers leurs propres pages et regroupé les sujets similaires.	N/A
7 novembre 2	Rubriques AMI mises à jour	Mise à jour de la rubrique AMI pour Slurm 23.11.10 et libjwt 17.0. Pour plus d'informations, consultez Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS et Étape 3 — Installation de Slurm. Simplification et correctio n des notes de publicati on pour AMIs. Pour de plus amples informations, veuillez consulter Notes de mise à jour pour un exemple de AWS PCS AMIs.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
7 novembre 2	Ajout d'une nouvelle rubrique sur l'utilisation de volumes EBS chiffrés avec PCS AWS	Ajout d'une rubrique qui décrit la politique de clé KMS requise pour les volumes EBS chiffrés dans AWS PCS. Pour de plus amples informations, veuillez consulter Politique de clé KMS requise pour une utilisation avec des volumes EBS chiffrés sur PCS AWS.	N/A
18 octobre 2024	AWS Publication de l'agent PCS 1.0.1-1	Documentation relative à l'AMI mise à jour pour faire référence à la version 1.0.1-1 de l'agent AWS PCS. Pour plus d’informations, consultez Des installateurs de logiciels à créer sur mesure AMIs pour les PC AWS et Étape 2 — Installation de l'agent AWS PCS.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
10 octobre 2(	Ajout d'un chapitre de résolution des problèmes	Ajout d'un chapitre de résolution des problèmes avec une rubrique sur le remplacement automatiq ue des instances EC2 après un redémarrage. Pour de plus amples informations, veuillez consulter Résolution des problèmes dans le service de calcul AWS parallèle.	N/A
23 septembre	Mise à jour des autorisat ions minimales pour utiliser les actions d'API et pour un administrateur de service	L'`ec2:Descr ibeInstan ceTypeOff erings` autorisation est désormais requise pour les actions `CreateCom puteNodeGroup` et `UpdateComputeNodeG roup` API. Pour de plus amples informati ons, veuillez consulter Autorisations minimales pour les AWS PCS.	N/A
5 septembre	Mise à jour de l'exemple de politique IAM concernant les autorisat ions minimales pour un administrateur de service	Pour de plus amples informations, veuillez consulter Autorisat ions minimales pour un administrateur de service.	N/A

Date	Modifier	Mises à jour de la documentation	Versions d'API mises à jour
5 septembre	Ajout d'une autorisation manquante au JSON dans la page des politiques gérées	Il s'agissait d'une correction apportée uniquement à la documentation. La politique gérée réelle n'a pas été modifiée. Pour de plus amples informations, veuillez consulter AWS politiques gérées pour le service de calcul AWS parallèle.	N/A
28 août 2024	Page de politiques gérées ajoutée	Pour de plus amples informations, veuillez consulter AWS politiques gérées pour le service de calcul AWS parallèle.	N/A
28 août 2024	AWS Version PCS	Première publication du guide de l'utilisateur du AWS PCS.	AWS SDK : 2024-08-28

AWS Glossaire

Pour la AWS terminologie la plus récente, consultez le AWS glossaire dans la Glossaire AWS référence.

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.